

日本国特許庁  
PATENT OFFICE  
JAPANESE GOVERNMENT

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出願年月日  
Date of Application: 2000年 6月 9日

出願番号  
Application Number: 特願2000-174573

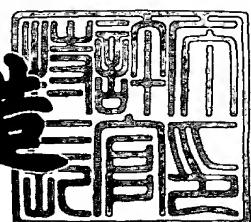
出願人  
Applicant(s): 任天堂株式会社



2000年10月27日

特許庁長官  
Commissioner,  
Patent Office

及川耕造



出証番号 出証特2000-3087984

【書類名】 特許願

【整理番号】 N000675

【あて先】 特許庁長官 殿

【国際特許分類】 A63F 13/00

## 【発明者】

【住所又は居所】 京都市東山区福稻上高松町60番地  
任天堂株式会社内

【氏名】 増山 巍

## 【発明者】

【住所又は居所】 京都市東山区福稻上高松町60番地  
任天堂株式会社内

【氏名】 鈴木 利明

## 【発明者】

【住所又は居所】 京都市東山区福稻上高松町60番地  
任天堂株式会社内

【氏名】 田原 明

## 【特許出願人】

【識別番号】 000233778

【氏名又は名称】 任天堂株式会社

【代表者】 山内 博

## 【先の出願に基づく優先権主張】

【出願番号】 平成11年特許願第282592号

【出願日】 平成11年10月 4日

## 【手数料の表示】

【予納台帳番号】 056085

【納付金額】 21,000円

## 【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

特2000-174573

【物件名】 要約書 1  
【フルーフの要否】 要

【書類名】 明細書

【発明の名称】 ゲームシステム及びそれに用いられるゲーム情報記憶媒体

【特許請求の範囲】

【請求項1】

ゲームプログラムを記憶するゲームプログラム記憶手段とゲームプログラムを実行する処理手段とを備えるゲーム装置に対して、処理手段の処理結果に基づく画像を表示する表示手段が関連的に設けられたゲームシステムであって、

プレイヤによって把持されるハウジング、および

前記ハウジングに関連して設けられ、かつハウジングに加えられた変化の量と方向の少なくとも一方を検出する変化状態検出手段を備え、

前記ゲームプログラム記憶手段は、

ゲームプレイ可能な空間を表示するための画像データを含むゲーム空間データと、

前記ゲーム空間データに基づいて前記表示手段にゲーム空間を表示させるための表示制御プログラムと、

前記変化状態検出手段の出力に基づいて、前記ハウジングに加えられた変化量と変化方向の少なくとも一方に関連させて、前記ゲーム空間の状態を変化させるようにシミュレートするシミュレーションプログラムとを記憶したことを特徴とする、ゲームシステム。

【請求項2】

前記変化状態検出手段は、変化量と変化方向として、前記ハウジングに加えられた傾きの傾き量と傾き方向の少なくとも一方を検出するものであり、

前記シミュレーションプログラムは、前記ハウジングに加えられた傾き量と傾き方向の少なくとも一方に関連させて、前記ゲーム空間が傾けられた状態となるようにシミュレートすることを特徴とする、請求項1に記載のゲームシステム。

【請求項3】

前記変化状態検出手段は、変化量と変化方向として、前記ハウジングに加えられた運動の運動量と運動方向の少なくとも一方を検出するものであり、

前記シミュレーションプログラムは、前記ハウジングに加えられた運動量と運

動方向の少なくとも一方に関連する運動状態になるように、前記ゲーム空間をシミュレートすることを特徴とする、請求項1に記載のゲームシステム。

【請求項4】

前記変化状態検出手段は、変化量と変化方向として、前記ハウジングに加えられた衝撃の衝撃量と衝撃方向の少なくとも一方を検出するものであり、

前記シミュレーションプログラムは、前記ハウジングに加えられた衝撃量と衝撃方向の少なくとも一方に関連させて、前記ゲーム空間が衝撃を加えられた状態となるようにシミュレートすることを特徴とする、請求項1に記載のゲームシステム。

【請求項5】

前記変化状態検出手段は、前記ハウジングに加えられた変化の量と方向の両方を検出するものであり、

前記シミュレーションプログラムは、前記ハウジングに加えられた変化量と変化方向の両方に関連させて、前記ゲーム空間の状態を変化させるようにシミュレートすることを特徴とする、請求項1ないし請求項4のいずれかに記載のゲームシステム。

【請求項6】

前記ハウジングは、前記ゲーム装置のハウジングであり、

前記ゲーム装置は、前記表示手段が前記ハウジングの一方主面に一体的に設けられた携帯ゲーム装置である、請求項1に記載のゲームシステム。

【請求項7】

前記ゲームプログラム記憶手段は、カートリッジに収納され、かつ前記携帯ゲーム装置の前記ハウジングに対して着脱自在に装着され、

前記変化状態検出手段は、前記カートリッジに収納され、当該カートリッジが前記携帯ゲーム装置の前記ハウジングに装着されたとき、携帯ゲーム装置のハウジングに加えられた変化量と変化方向の少なくとも一方を検出する、請求項6に記載のゲームシステム。

【請求項8】

前記変化状態検出手段は、前記携帯ゲーム機のハウジングに加えられる変化状

態によって道具として操作したことを検出するものであり、

前記ゲームプログラム記憶手段は、前記ゲーム空間上を移動可能な移動キャラクタを表示するためのキャラクタデータ記憶部を含み、

前記ゲーム空間データは、前記ゲーム空間上に表示される前記移動キャラクタの動きを制御する機能を有する道具を連想させる表示を行うための画像データであり、

前記ゲームプログラム記憶手段は、前記キャラクタデータ記憶部に記憶されている移動キャラクタを読み出し、前記変化状態検出手段の出力に基づいて、前記ハウジングに加えられた変化量と変化方向の少なくとも一方に関連させて、移動キャラクタの表示状態が前記道具によって制御されているように処理するキャラクタ制御プログラムを含む、請求項7に記載のゲームシステム。

#### 【請求項9】

前記ゲームプログラム記憶手段は、前記ゲーム空間上を移動可能な移動キャラクタを表示するためのキャラクタデータ記憶部を含み、

前記ゲームプログラム記憶手段は、前記キャラクタデータ記憶部に記憶されている移動キャラクタを読み出し、前記変化状態検出手段の出力に基づいて、前記ハウジングに加えられた変化量と変化方向の少なくとも一方に関連させて、移動キャラクタの表示状態を変化させるように制御するキャラクタ制御プログラムを含む、請求項1に記載のゲームシステム。

#### 【請求項10】

前記ゲームプログラム記憶手段は、プレイヤの操作とは無関係にゲーム空間上を予め定められたプログラムに従って第1の動きをするノンプレイヤキャラクタを表示するためのノンプレイヤキャラクタデータ記憶部とを含み、

前記シミュレーションプログラムは、

前記変化状態検出手段によって変化量と変化方向のいずれの変化状態が検出されないとき、前記ノンプレイヤキャラクタがプログラムによって予め定められた第1の動きをするように制御し、

前記変化状態検出手段によって変化量と変化方向の少なくとも一方の変化状態が検出されたないとき前記ノンプレイヤキャラクタが第1の動きに加えて変化

状態検出手段の出力に基づく変化量と変化方向の少なくとも一方に関連した第2の動きをするように制御することを特徴とする、請求項1に記載のゲームシステム。

#### 【請求項11】

前記ゲームプログラム記憶手段は、前記ゲーム空間上を移動可能な移動キャラクタを表示するためのキャラクタデータ記憶部を含み、

前記ゲーム空間データは、前記移動キャラクタがゲーム空間上を移動するときに、移動キャラクタの動きが他の領域よりも異なるように定められた特定領域を表示するためのデータを含み、

前記シミュレーションプログラムは、前記変化状態検出手段の出力に基づいて前記ハウジングに加えられた変化量と変化方向の少なくとも一方に関連させて、前記移動キャラクタの表示状態を制御し、移動キャラクタが前記特定領域に存在するときに、移動キャラクタの動きを他の領域に存在していたときとは異なれるように表示制御することを特徴とする、請求項1に記載のゲームシステム。

#### 【請求項12】

前記ゲーム空間データは、前記表示手段によって表示可能な表示領域よりも大きなゲーム空間を表示するための空間データを含み、

前記表示制御プログラムは、前記ゲーム空間のうちの表示領域の範囲内にある一部のゲーム空間の画像データを前記表示手段に表示させるものであり、

前記シミュレーションプログラムは、前記変化状態検出手段の出力の変化量と変化方向の少なくとも一方に基づいて表示領域内に存在するゲーム空間のみの状態を変化させるようにシミュレートすることを特徴とする、請求項1に記載のゲームシステム。

#### 【請求項13】

前記変化状態検出手段は、変化量として前記ハウジングの移動量を検出し、かつ変化方向として移動方向を検出するものであり、

前記ゲーム空間データは、前記表示手段の表示領域よりも大きなゲーム空間を表示するための空間データを含み、

前記表示制御プログラムは、表示領域に対応するゲーム空間の一部の空間領域

を前記表示手段に表示させ、前記ハウジングの移動に応じて、ゲーム空間の表示領域を移動方向へ移動量に相当する領域だけ徐々に移動させることを特徴とする、請求項1に記載のゲームシステム。

#### 【請求項14】

前記ゲーム装置は、プレイヤによって操作される操作手段を前記ハウジングの一方主面に備え、

前記シミュレーションプログラムは、前記変化状態検出手段の検出出力と前記操作手段の操作状態に基づいて、ゲーム空間の状態をシミュレートするように変化させることを特徴とする、請求項1に記載のゲームシステム。

#### 【請求項15】

前記ゲームプログラム記憶手段は、前記シミュレーションプログラムに基づく前記ゲーム空間の状態に対応した音を発生する音発生プログラムを含む、請求項1に記載のゲームシステム。

#### 【請求項16】

表示手段が関連的に設けられ、プレイヤによって把持されるハウジングを含む操作手段と、ハウジングに関連して設けられかつハウジングに加えられた変化の量と方向の少なくとも一方を検出する変化状態検出手段と、プログラムを処理することによって得られる画像を表示手段に表示させる処理手段とから構成されるゲームシステムに対して、着脱自在に装着されかつゲームプログラムを記憶するゲーム情報記憶媒体であって、

ゲームプレイ可能な空間を表示するための画像データを含むゲーム空間データと、

前記ゲーム空間データに基づいて前記表示手段にゲーム空間を表示させるための表示制御プログラムと、

前記変化状態検出手段の出力に基づいて、前記ハウジングに加えられた変化量と変化方向の少なくとも一方に関連させて、前記ゲーム空間の状態を変化させるようにシミュレートするシミュレーションプログラムとを記憶したことを特徴とする、ゲーム情報記憶媒体。

#### 【請求項17】

表示手段が一体的に設けられかつプレイヤによって把持されるハウジングと、プログラムを処理することによって得られる画像を表示手段に表示させる処理手段とを含む携帯ゲーム装置に対して、着脱自在に装着されかつゲームプログラムを記憶するゲーム情報記憶媒体であって、

前記ゲーム情報記憶媒体に関連して設けられ、かつ前記携帯ゲーム装置のハウジングに加えられた変化の量と方向の少なくとも一方を検出する変化状態検出手段を備え、

ゲームプレイ可能な空間を表示するための画像データを含むゲーム空間データと、

前記ゲーム空間データに基づいて前記表示手段にゲーム空間を表示させるための表示制御プログラムと、

前記変化状態記検出手段の出力に基づいて、前記ハウジングに加えられた変化量と変化方向の少なくとも一方に関連させて、前記ゲーム空間の状態を変化させるようにシミュレートするシミュレーションプログラムとを記憶したことを特徴とする、ゲーム情報記憶媒体。

#### 【請求項18】

前記変化状態検出手段は、前記ハウジングに加えられた変化の量と方向の両方を検出するものであり、

前記シミュレーションプログラムは、前記ハウジングに加えられた変化量と変化方向の両方に関連させて、前記ゲーム空間の状態を変化させるようにシミュレートすることを特徴とする、請求項17に記載のゲーム情報記憶媒体。

#### 【請求項19】

少なくとも2台のゲーム装置によって構成されるゲームシステムであって、

前記2台のゲーム装置は、それぞれゲームプログラムを記憶するゲームプログラム記憶手段と、ゲームプログラムを実行する処理手段と、プレイヤによって把持されるハウジングとを備え、処理手段の処理結果に基づく画像を表示する表示手段が関連的に設けられ、

前記2台のゲーム装置の少なくとも一方は、前記ハウジングに関連して設けられ、かつハウジングに加えられた変化量と変化方向の少なくとも一方を検出する

変化状態検出手段を備え、

前記2台のゲーム装置に接続されかつ相互に関連するデータを相手側のゲーム装置に伝送するデータ伝送手段をさらに備え、

前記2台のゲーム装置のそれぞれの前記ゲームプログラム記憶手段は、

ゲームプレイ可能な空間を表示するための画像データを含むゲーム空間データと、

前記ゲーム空間データに基づいて前記表示手段にゲーム空間を表示させるための表示制御プログラムとを記憶し、

前記2台のゲーム装置の少なくとも他方の前記ゲームプログラム記憶手段は、前記データ伝送手段を介して伝送される前記一方のゲーム装置の前記変化状態検出手段の出力に基づいて、一方のゲーム装置の前記ハウジングに加えられた変化量と変化方向の少なくとも1つに関連させて、他方のゲーム装置の前記ゲーム空間の状態を変化させるようにシミュレートするシミュレーションプログラムをさらに記憶したことを特徴とする、ゲームシステム。

#### 【請求項20】

前記変化状態検出手段は、2台のゲーム装置のそれぞれに備えられ、

前記2台のゲーム装置のそれぞれの前記ゲームプログラム記憶手段は、一方のゲーム装置の前記変化状態検出手段の出力に基づいて、一方のゲーム装置の前記ハウジングに加えられた変化量と変化方向の少なくとも1つに関連させて、他方のゲーム装置の前記ゲーム空間の状態を変化させるようにシミュレートするシミュレーションプログラムを記憶したことを特徴とする、請求項19に記載のゲームシステム。

#### 【請求項21】

前記一方のゲーム装置の前記ゲームプログラム記憶手段に記憶されるゲーム空間データと、前記他方のゲーム装置の前記ゲームプログラム記憶手段に記憶されるゲーム空間データとは、同一のゲーム空間データに選ばれ、

前記一方のゲーム装置の前記シミュレーションプログラムは、当該ゲーム空間制御プログラムによってシミュレートされる他方のゲーム空間の状態に対応して、前記一方のゲーム装置の前記ゲーム空間の状態を変化させ、

前記他方のゲーム装置の前記シミュレーションプログラムは、当該ゲーム空間制御プログラムによってシミュレートされる一方のゲーム空間の状態に対応して、前記他方のゲーム装置の前記ゲーム空間の状態を変化させることを特徴とする、請求項19に記載のゲームシステム。

【発明の詳細な説明】

【0001】

【産業上の利用分野】

この発明はゲームシステム及びそれに用いられるゲーム情報記憶媒体に関し、特に携帯ゲーム装置のハウジング又はビデオゲーム装置のコントローラに加えられる傾き、運動又は衝撃等の変化量・変化方向を検出して操作情報としてゲーム空間の状態を変化させる、ゲームシステム及びそれに用いられるゲーム情報記憶媒体に関する。

【0002】

【従来技術】

従来のゲーム装置における操作は、プレイヤがビデオゲーム機のコントローラ（操作器のハウジング）又は携帯型ゲーム装置のハウジングを両手で把持しつつ、方向指示キー（ジョイスティック）やボタン等の操作スイッチを操作することによって行われていた。例えば、プレイヤが方向指示キーの上下左右の何れか1つの押点を押圧すると、移動（プレイヤ）キャラクタが上下左右の何れか1つの押圧方向に移動し、動作ボタンを操作すると移動キャラクタがジャンプ等の動作ボタンに定義付けされた動作をするように、移動キャラクタの表示状態が変化する。

また、従来のゲーム装置又はゲームソフト（ゲーム情報記憶媒体）は、プレイヤが操作スイッチを操作してプレイヤの分身である移動（プレイヤ）キャラクタの画面上の動きを変化させるものであるため、ゲーム空間（又は背景画面）をプレイヤの操作によって自由に変更できるものではなかった。

【0003】

【発明が解決しようとする課題】

従来のゲーム操作方法は、プレイヤがゲームソフトの取扱説明書等に示された

ゲーム操作方法を覚えることが必要であり、汎用的な操作スイッチを使用するものであるため、プレイヤによる操作スイッチの操作感覚と一致したゲーム空間（又はゲーム画面）の変化を実現できず、操作感覚と画面の表示状態とが対応しない場合もあった。そのため、プレイヤはゲーム操作方法を習得するまでは、ゲームプレイに集中できず、興趣を損なうこともあった。

また、従来のゲーム装置又はゲームソフトは、プレイヤの操作に応じてゲーム空間（又は背景画面）を変化させることができず、ゲーム空間の変化に乏しく、面白味に欠ける場合もあった。

#### 【0004】

それゆえに、この発明の主たる目的は、ゲーム空間の状態をプレイヤの操作によって変化させることができる、ゲームシステム及びそれに用いられるゲーム情報記憶媒体を提供することである。

この発明の他の目的は、簡単な操作でゲーム空間の状態を変化させることができあり、プレイヤが操作方法に熟練する必要がなく、ゲームプレイに集中できるとともに、熱中度を高められる、ゲームシステム及びそれに用いられるゲーム情報記憶媒体を提供することである。

この発明のその他の目的は、プレイヤの操作とゲーム空間の状態の変化を対応させて、操作感覚と対応したゲーム画面の変化を実現し得る、ゲームシステム及びそれ用いられるゲーム情報記憶媒体を提供することである。

この発明のさらに他の目的は、複数のゲーム装置と連動してゲーム空間の状態を変化でき、複数のプレイヤによる協力又は対戦によってゲーム空間の変化状態の種類を豊富にでき、ゲームの興趣を高められて、仮想現実感を味わうことができる、ゲームシステム及びそれ用いられるゲーム情報記憶媒体を提供することである。

#### 【0005】

##### 【課題を解決するための手段】

第1の発明（請求項1に係る発明）は、ゲームプログラムを記憶するゲームプログラム記憶手段とゲームプログラムを実行する処理手段とを備えるゲーム装置に対して、処理手段の処理結果に基づく画像を表示する表示手段が関連的に設け

られたゲームシステムである。そして、プレイヤによって把持されるハウジング、及び変化状態検出手段を備える。変化状態検出手段は、ハウジングに関連して設けられ、かつハウジングに加えられた変化の量（例えば、傾き量、運動量、衝撃量等）と変化方向（例えば、傾き方向、運動方向、衝撃方向等）の少なくとも一方を検出する。ゲームプログラム記憶手段は、ゲーム空間データと、表示制御プログラムと、シミュレーションプログラムとを記憶する。

ゲーム空間データは、ゲームプレイ可能な空間を表示するための画像データを含む。表示制御プログラムは、ゲーム空間データに基づいて表示手段にゲーム空間を表示させる。シミュレーションプログラムは、状態検出手段の出力に基づいて、ハウジングに加えられた変化量と変化方向の少なくとも一方に関連させて、ゲーム空間の状態を変化させるようにシミュレートする。

#### 【0006】

ここで、ゲーム空間とは、プレイ可能なゲーム中の世界を意味するものであり、ゲームの種類又はジャンルによって異なり、表示画面を通じてプレイヤに示されるものである。例えば、移動（又はプレイヤ）キャラクタがその中を移動するアクションゲームやロールプレイングゲームでは背景、迷路その他のマップ、格闘ゲームではリング（それに加えて観客席やリング上部の空間も含む）、レースゲームでは走行可能なコースやコースの周辺の空間、シューティングゲームではキャラクタの背景となるような宇宙空間等の背景画面（但し、キャラクタは必須ではなく、キャラクタが存在しないゲーム空間も考えられ得る）、道具を使用したゲームでは道具の使用を連想させるような画面等がゲーム空間となる。

シミュレートとは、ハウジングに加えられた傾き、運動又は衝撃の量的なものと方向性の少なくとも一方に基づいて、実空間上に生じる変化をゲーム空間の状態変化として擬似的に表現するゲーム制御のことである。ゲーム制御には、ゲーム空間そのものの状態変化をシミュレートする場合と、ゲーム空間の状態が変化することによって他の物体への間接的な影響をシミュレートする場合を含む。前者は、ハウジングに衝撃が与えられたとき、ゲーム空間にエネルギーが与えられたものと想定してゲーム空間の地形を変形させるようにシミュレートする場合である。後者は、ハウジングが傾けられたとき、ゲーム空間の一例の迷路板が傾い

たものと仮定として、迷路板の上に存在するボールが転がるようにシミュレートする場合である。ゲーム空間そのものの状態変化をシミュレートする場合は、地形変形を引き起こすもののような表示上の変化を引き起こすもののに他に、ゲーム空間の温度上昇等のパラメータが変化するものも考えられる。

#### 【0007】

第2の発明（請求項16に係る発明）は、表示手段が関連的に設けられ、プレイヤによって把持されるハウジングを含む操作手段と、ハウジングに関連して設けられかつハウジングに加えられた変化の量と方向の少なくとも一方を検出する変化状態検出手段と、プログラムを処理することによって得られる画像を表示手段に表示させる処理手段とから構成されるゲームシステムに対して、着脱自在に装着されかつゲームプログラムを記憶するゲーム情報記憶媒体である。ゲーム情報記憶媒体は、ゲーム空間データと、表示制御プログラムと、シミュレーションプログラムとを記憶する。

ゲーム空間データは、ゲームプレイ可能な空間を表示するための画像データを含む。表示制御プログラムは、ゲーム空間データに基づいて表示手段にゲーム空間を表示させるものである。シミュレーションプログラムは、変化状態検出手段の出力に基づいて、ハウジングに加えられた変化量と変化方向の少なくとも一方に関連させて、ゲーム空間の状態を変化させるようにシミュレートするものである。

#### 【0008】

第3の発明（請求項17に係る発明）は、表示手段が一体的に設けられかつプレイヤによって把持されるハウジングと、プログラムを処理することによって得られる画像を表示手段に表示させる処理手段とを含む携帯ゲーム装置に対して、着脱自在に装着されかつゲームプログラムを記憶するゲーム情報記憶媒体であり、ゲーム情報記憶媒体に関連して設けられ、かつ携帯ゲーム装置のハウジングに加えられた変化の量と方向の少なくとも一方を検出する変化状態検出手段を備える。

ゲーム情報記憶媒体は、ゲーム空間データと、表示制御プログラムと、シミュレーションプログラムとを記憶する。ゲーム空間データは、ゲームプレイ可能な

空間を表示するための画像データを含む。表示制御プログラムは、ゲーム空間データに基づいて前記表示手段にゲーム空間を表示させるプログラムである。シミュレーションプログラムは、変化状態検出手段の出力に基づいて、ハウジングに加えられた変化量と変化方向の少なくとも一方に関連させて、ゲーム空間の状態を変化させるようにシミュレートするプログラムである。

## 【0009】

第4の発明（請求項19に係る発明）は、少なくとも2台のゲーム装置によって構成されるゲームシステムであって、複数のゲーム装置の連動を行うものである。2台のゲーム装置は、それぞれゲームプログラムを記憶するゲームプログラム記憶手段と、ゲームプログラムを実行する処理手段と、プレイヤによって把持されるハウジングとを備え、処理手段の処理結果に基づく画像を表示する表示手段が関連的に設けられる。2台のゲーム装置の少なくとも一方は、ハウジングに関連して設けられ、かつハウジングに加えられた変化量と変化方向の少なくとも一方を検出する変化状態検出手段を備える。2台のゲーム装置に接続されかつ相互に関連するデータを相手側のゲーム装置に伝送するデータ伝送手段をさらに備える。

2台のゲーム装置のそれぞれのゲームプログラム記憶手段は、ゲーム空間データと、表示制御プログラムとを記憶する。ゲーム空間データは、ゲームプレイ可能な空間を表示するための画像データを含む。表示制御プログラムは、ゲーム空間データに基づいて表示手段にゲーム空間を表示させるプログラムである。2台のゲーム装置の少なくとも他方のゲームプログラム記憶手段は、データ伝送手段を介して伝送される一方のゲーム装置の変化状態検出手段の出力に基づいて、一方のゲーム装置の前記ハウジングに加えられた変化量と変化方向の少なくとも1つに関連させて、他方のゲーム装置の前記ゲーム空間の状態を変化させるようにシミュレートするシミュレーションプログラムをさらに記憶する。

## 【0010】

## 【発明の効果】

この発明によれば、ゲーム空間の状態を変化させることが可能な、ゲームシステム及びそれ用いられるゲーム情報記憶媒体が得られる。

また、この発明によれば、簡単な操作でゲーム空間の状態を変化させることができ、プレイヤが操作方法に熟練する必要がなく、ゲームプレイに集中でき、熱中度を高められる、ゲームシステム及びそれに用いられるゲーム情報記憶媒体が得られる。

また、この発明によれば、プレイヤの操作とゲーム空間の状態の変化を対応させて、操作感覚と対応したゲーム画面の変化を実現することができる、ゲームシステム及びそれ用いられるゲーム情報記憶媒体が得られる。

さらに、この発明によれば、複数のゲーム装置と連動してゲーム空間の状態を変化でき、複数のプレイヤによる協力又は対戦によってゲーム空間の変化状態の種類を豊富にでき、ゲームの興味を高められて、仮想現実感を味わうことができる、ゲームシステムが得られる。

#### 【0011】

##### 【発明の実施の形態】

###### (第1実施例)

図1～図40を参照して、本発明の第1の実施例の携帯型ゲーム装置を説明する。図1は携帯型ゲーム装置（以下、単に「ゲーム装置」と呼ぶ）の外観図である。ゲーム装置は、ゲーム装置本体10とゲーム装置本体10に着脱自在なゲームカートリッジ（以下「カートリッジ」と略称する）30とから構成される。カートリッジ30は、ゲーム装置本体10に装着されたときに電気的に接続される。ゲーム装置本体10は、ハウジング11を備え、ハウジング11の内部に後述する図3に示すように回路構成される基板を含む。ハウジング11の一方主面にはLCD12および操作スイッチ13a～13eが設けられ、他方主面にはカートリッジ30を装着するための孔14が形成される。また側面には必要に応じて他のゲーム装置と通信するための通信ケーブルを接続するコネクタ15が設けられる。

#### 【0012】

図2は、ゲーム装置とXYZ軸との関係を示した図である。LCD12を上方に向けて操作スイッチ部が手前になるようにゲーム装置を配置したとき、ゲーム装置の横方向をX軸（右方向がプラス方向）とし、縦方向をY軸（奥方向がプラ

ス方向) とし、厚み方向をZ軸(上方向がプラス方向) とする。

#### 【0013】

図3はゲーム装置のブロック図である。ゲーム装置本体10は、基板27を内蔵し、基板27にはCPU21が実装される。CPU21には、LCDドライバ22と操作スイッチ13とサウンド発生回路23と通信用インターフェース24と表示用RAM25とワークRAM26とが接続される。サウンド発生回路23には、スピーカー16が接続される。通信用インターフェース24は、コネクタ15および通信ケーブル50を介して他のゲーム装置40と接続される。なお、他のゲーム装置40との通信の方法は通信ケーブル50による方法を図示したが、無線や携帯電話等を使用したものでも良い。

#### 【0014】

カートリッジ30は、基板36を内蔵し、基板36には図16を参照して後述するようなゲームプログラムおよびゲームデータを記憶したプログラムROM34と、図19を参照して後述するようなゲームデータを記憶するバックアップRAM35が実装される。カートリッジ30は、これらの記憶手段に加えて、ゲーム装置本体の傾き、運動および衝撃を検出する検出手段の一例として、X軸方向およびY軸方向の加速度を検出するXY軸加速度センサ31と、Z軸方向の加速度を検出するZ軸接点スイッチ32を含む。また、カートリッジ30は、加速度検出手段のインターフェースであるセンサインターフェース33を含む。X軸、Y軸、Z軸方向すべての加速度を検出可能な3軸の加速度センサを使用する場合、Z軸接点スイッチ32が不要となるが、2軸加速度センサ(XY軸加速度センサ)の方が安価であり、本実施例ではZ軸方向の加速度検出は高い精度を必要としないので、構造が簡単で安価なZ軸接点スイッチ32を用いることにする。また、XY軸方向についても高い精度を必要としない場合は、Z軸接点スイッチと同じような構造の検出手段をXY軸方向の加速度検出に使用しても良い。

#### 【0015】

プログラムROM34に記憶されたゲームプログラムはCPU21によって実行される。ゲームプログラムの実行の際に必要な一時的なデータがワークRAM26に記憶される。ゲーム装置の電源をオフしたときにも持続的に記憶すべきゲ

ームデータがバックアップRAM35に記憶される。CPU21がゲームプログラムを実行して得た表示データが表示用RAM25に記憶され、LCDドライバ22を介してLCD12に表示される。同様に、CPU21がゲームプログラムを実行して得たサウンドデータがサウンド発生回路23に送られてスピーカー16から効果音としてサウンドが発生される。操作スイッチ13はゲーム操作をするためのものであるが、本実施例においては操作スイッチ13は補助的なものである。プレイヤは、主として、ゲーム装置を傾けたり、運動させたり、衝撃を与えてたりしてゲーム操作をおこなう。このゲーム装置の傾き、運動、衝撃のゲーム操作がXY軸加速度センサ31およびZ軸接点スイッチ32によって検出される。これらの加速度検出手段の出力値を利用して、CPU21がゲームプログラムを実行する。

#### 【0016】

複数のゲーム装置を使用するゲームの場合、CPU21がゲームプログラムを実行して得たデータは通信用インターフェース24に送られてコネクタ15および通信ケーブル50を介して他のゲーム装置40に送られる。また、他のゲーム装置40のデータが通信ケーブル50、コネクタ15および通信用インターフェース24を介してCPU21に送られる。

#### 【0017】

図4はセンサインターフェース33の詳細ブロック図である。センサインターフェース33は、Xカウンタ331、Yカウンタ332、カウント停止回路333、ラッチ334と335、デコーダ336および汎用I/Oポート337を含む。Xカウンタ331は、XY軸加速度センサ31のX軸出力に基づいてクロック信号Φのパルスをカウントする。Yカウンタ332は、Y軸出力に基づいてクロック信号Φのパルスをカウントする。カウント停止回路333は、XY軸加速度センサ31のX軸出力の立ち下がりに応答して、Xカウンタ331にカウント停止信号を送り、Y軸出力の立ち下がりに応答してYカウンタ332にカウント停止信号を送る。ラッチ334、335は、Xカウンタ331、Yカウンタ332の値をそれぞれ保持する。デコーダ336は、Xカウンタ331とYカウンタ332とラッチ334とラッチ335にスタート/リセット信号を送信する。汎

用I/Oポート337は、拡張ユニットを接続するために用いられる。ラッチ334, 335はZ軸接点スイッチの出力値（0または1）も保持する。具体的にはラッチ334, 335の最上位ビットがZ軸接点スイッチの出力値に割り当てられ、残りの下位ビットがXカウンタまたはYカウンタの値に割り当てられる。汎用I/Oポート337に接続される拡張ユニットは、例えばゲームに現実感を与えるためにゲームプログラムに連動して振動するような振動ユニット等がある。

## 【0018】

図5はセンサインターフェース33がXY軸加速度センサ31の出力から加速度の大きさに応じたカウント値を計測する原理を示した図である。本実施例におけるXY軸加速度センサ31は、波形の1周期（期間1）のうちのデューティー比を変化させることによって加速度の大きさを表す信号を出力する。この場合、1周期のうちのハイレベル期間（期間2または期間3）の比率が大きいほど大きな加速度を検出したことを示す。また、XY軸加速度センサ31は、X軸出力からX軸方向の加速度の大きさを出力し、Y軸出力からY軸方向の加速度の大きさを出力する。

## 【0019】

Xカウンタ331は、デコーダ336から出力されるカウントスタート信号がローレベルになると、X軸出力のローレベルからハイレベルへの立ち上がりを検出した後、カウント動作を開始する。具体的には、Xカウンタ331はクロック信号Φが与えられる毎にそのカウント値を歩進し、カウント停止回路333からのカウント停止信号に応じてカウント動作を停止する。このようにして、Xカウンタ331はカウントスタート信号がローレベルになった直後のX軸出力のハイレベルへの立ち上がりからローレベルへの立ち下がりまでの期間（期間2）中に、クロック信号Φをカウントする。Yカウンタ332も同様に、カウントスタート信号がローレベルになった直後のY軸出力のハイレベルへの立ち上がりからローレベルへの立ち下がりまでの期間（期間3）中に、クロック信号Φをカウントする。このようにして、Xカウンタ331はX軸方向の加速度の大きさに応じたカウント値を保持し、Yカウンタ332はY軸方向の加速度の大きさに応じたカウ

ント値を保持する。Xカウンタ331およびYカウンタ332の値はラッチ334またはラッチ335に保持され、ラッチ334およびラッチ335のデータは、データバスを介してCPU21に読み出されてゲームプログラムで利用される。

#### 【0020】

Xカウンタ331およびYカウンタ332は、例えば、0から31までカウントし、例えば、カウント値15を基準（加速度0）として、カウント値が0のときが-2G（マイナス方向に重力加速度の2倍）、カウント値が31のときが2G（プラス方向に重力加速度の2倍）であるように設定される。CPU21は、ゲームプログラムに基づいてこれらのカウント値を読み込むが、このとき、カウント値15を0、カウント値0を-15、カウント値31を16として読み込むので、XY軸加速度センサ31がマイナス方向の加速度を検出したときはCPUの読み込み値はマイナスであり、プラス方向の加速度を検出したときにはCPUの読み込み値はプラスとなる。

#### 【0021】

図6はZ軸接点スイッチ32の構造図である。Z軸接点スイッチ32は導体よりなる球接点321と接点322と接点323と箱体324から構成される。具体的には、球接点321は、箱体324の空間内のほぼ中央部で移動自在に支持される（箱体324の内面底部には球接点321をほぼ中央に支持するためのくぼみ（324a）が設けられる）。箱体324の上部には、それぞれの一方端部に半円状の切欠部（322a, 323a）を有する板状の接点322と接点323とが一方端部を対向させた状態で、他方端部が基板36に固着される。また、箱体324は、接点322および接点323によって吊り下げられた状態で、基板36に固定的に保持される。このような構成によって、カートリッジ30がZ軸方向（プラス方向またはマイナス方向）へ勢いよく移動されると、図7に示すように球接点321が箱体324内でZ軸方向へ移動して接点322と接点323にほぼ同時に接触して、接点322と接点323が球接点321を介して導通状態となり、Z軸方向への加速度入力のあったことが検出される。接点322と接点323の導通時間に基づいてZ軸方向への加速度の大小が検出されることに

なる。なお、カートリッジ30が穏やかに傾けられた場合は、球接点321が箱体324内を移動するが接点322と接点323を短絡しないので、加速度を検出しない。

## 【0022】

図8はゲーム画面の一例である。ゲーム画面には、プレイヤキャラクタの一例のボール61と、敵キャラクタ（以下「N P C」と略す）の一例の亀62と、迷路を構成する壁63や穴64とが表示される。ゲームマップは、LCD12の表示範囲より広い仮想マップであるので、LCD12には、ゲームマップの一部領域のみが表示され、プレイヤキャラクタの移動に伴ってスクロールする。また、LCD12には亀62a～62cの3匹のみ表示されているが、ゲームマップ上には他にも多数の亀が存在する。また、ゲームマップには、床面と氷面と水中等の地形が存在する。

## 【0023】

ボール61は、プレイヤがゲーム装置を傾けたり、運動や衝撃を与えるように操作することによって、その移動量や移動方向が変更され、必要に応じてその形状も変化される。亀62は、ゲームプログラムによって移動制御（自律移動）されるが、プレイヤがゲーム装置を傾けたり、運動や衝撃を与えた場合にも移動したり表示変化する。

## 【0024】

このゲームの概要を説明すると、プレイヤーは壁63によって迷路状になっているゲームマップの中でボール61を操作して、N P Cである亀62a～62cをボール61で押しつぶす。押しつぶされた亀は消滅される。ゲームマップ上の全ての亀を消滅させることに成功するとゲームクリアとなる。ゲームマップ上にはいくつかの穴64があり、この穴64にボール61が落ちると1回のミスとなるか、またはゲームオーバーとなる。

## 【0025】

図9から図12はゲーム操作の例を示した図である。図9はX軸またはY軸方向のスライド入力を示した図である。X軸方向の移動（スライド）がXY軸加速度センサ31のX軸出力に基づいて検出され、Y軸方向の移動（スライド）がX

Y軸加速度センサ31のY軸出力に基づいて検出される。（X軸またはY軸方向に移動させると、加速度が生じる）。図10はX軸またはY軸を中心とした傾き入力を示した図である。X軸を中心とした傾きがXY軸加速度センサ31のY軸出力に基づいて検出され、Y軸を中心として傾きがXY軸加速度センサ31のX軸出力に基づいて検出される（X軸を中心として傾きが生じると、重力によってY軸方向に加速度が生じ、Y軸を中心として傾きが生じると、重力によってX軸方向に加速度が生じる）。図11はX軸方向またはY軸方向の衝撃入力を示した図である。X軸方向の加速度入力がXY軸加速度センサ31のX軸出力から出力されるが、この出力値が一定値以上の場合、衝撃入力があったとする。また、Y軸方向の加速度入力がXY軸加速度センサ31のY軸出力から出力されるが、この出力値が一定値以上の場合、衝撃入力があったとする。図12はZ軸方向の運動入力（または衝撃入力）を示した図である。Z軸方向の運動（または衝撃）がZ軸接点スイッチ32によって検出される。

## 【0026】

図13乃至図15は前述のゲーム操作それぞれについて利用方法の例を示した図である。図13はスライド入力の利用方法を示した図である（図30を参照して後述するゲームマップ選択処理におけるゲーム画面の一例でもある）。LCD12の表示範囲より大きな仮想マップの一部領域をLCD12に表示している場合に、スライド入力することによって、表示領域をスクロールさせる。具体的には、X軸の+方向にスライド入力した場合は、現在の表示領域からX軸の+方向に移動した領域を表示するようとする。Y軸方向のスライド入力も同様に処理する。スライド入力をこのように処理することによって、あたかもLCD12を通して広い世界の一部を覗いているような感覚をプレイヤに与えることができる。なお、本実施例においては、このスライド入力は、図30を参照して後述するゲームマップ選択処理において利用するのみであり、ゲームのメインの処理におけるゲームマップのスクロール処理においては、スライド入力は利用しない。ゲームマップのスクロール処理の方法については、図38乃至図40を参照して後述する。

## 【0027】

図14はX軸またはY軸を中心とした傾き入力の利用方法を示した図である。X軸を中心とした傾き入力があったときは、ゲーム画面上のゲームキャラクタ（プレイヤーキャラクタ61およびNPC62）がY軸方向に平行移動するような表示をする（X軸を中心としてプラス方向に傾いている場合には、ゲームキャラクタがY軸のマイナス方向に平行移動するような表示をする）。また、Y軸を中心とした傾き入力があったときは、ゲーム画面上のゲームキャラクタ（プレイヤーキャラクタ61およびNPC62）がX軸方向に平行移動するような表示をする（Y軸を中心としてマイナス方向に傾いている場合には、ゲームキャラクタがX軸のマイナス方向に平行移動するような表示をする）。傾き入力をこのように処理することによって、ゲーム空間である迷路板がゲーム装置と同じように傾いて、傾いた迷路板上をゲームキャラクタが滑っている（転がっている）かのような感覚をプレイヤに与えることができる。なお、ゲームマップには、床面、水面、水中等のように、ボール61の移動量を変化させる要因となる地形が混在しており、ゲームキャラクタがどこに存在しているかによって傾き入力に応じた移動量が変化する。例えば、水面の場合は滑りやすいので移動量が大きく、水中の場合は移動量が少ないように、ボール61の制御の大きさを変化させる。

## 【0028】

図15は衝撃入力またはZ軸方向の運動入力の利用方法を示した図である。X軸方向またはY軸方向の衝撃入力があったとき、傾き入力の処理（迷路板の傾きによるゲームキャラクタの移動）とは異なった処理を行う。例えば、ゲーム空間である水に波を起こす。X軸のプラス方向の衝撃入力があった場合、X軸のプラス方向に波を発生する。X軸のマイナス方向の衝撃入力があった場合、X軸のマイナス方向に波を発生する。Y軸方向の衝撃入力についても同様である。また、X軸方向の加速度入力をX軸方向のベクトル成分とし、Y軸方向の加速度入力をY軸方向のベクトル成分として合成したベクトルの方向に波を発生しても良い。この波によってゲームキャラクタが流されて移動するような表示をする。波に流れられている間は、ゲームキャラクタが制御不可能であるようにしてもよい。また、Z軸方向の運動入力（または衝撃入力）があったとき、プレイヤーキャラクタであるボール61がジャンプするように表示変化される。Z軸方向の運動入力をこ

のように処理することによって、ゲーム空間である迷路板がゲーム装置と同じようにZ軸方向に運動して、迷路板上のゲームキャラクタをジャンプさせるかのような感覚をプレイヤに与えることができる。ジャンプしている間は、傾き入力があったときでもボール61が移動しない。また、Z軸方向の運動入力（または衝撃入力）があったとき、NPCである亀62は裏返る（裏返っていた亀は表向きに戻る）。亀は裏返ると滑りやすくなり、表向きの場合に比べて傾き入力があったときの移動量が大きいように移動処理される。

## 【0029】

図16は、プログラムROM34のメモリマップである。プログラムROM34には、CPU21によって実行されるゲームプログラムおよびゲームデータが記憶される。プログラムROM34は、具体的には、オブジェクトキャラクタデータ記憶領域34a、マップデータ記憶領域34b、加速度センサ出力値変換テーブル記憶領域34cおよびゲームプログラム記憶領域34eを含む。オブジェクトキャラクタデータ記憶領域34aには、オブジェクトキャラクタのグラフィックデータが記憶される。オブジェクトキャラクタはいくつかのポーズを持っている（例えばNPCである亀の「表向き」と「裏向き」）、一つのキャラクタ毎にポーズに応じた複数のグラフィックデータが記憶されている。マップデータ記憶領域34bには、ゲームマップ毎のマップデータおよびゲームマップ選択用マップが記憶されている。ゲームマップ選択用マップは、図30を参照して後述するゲームマップ選択処理においてLCD12に表示される仮想マップのデータである。

## 【0030】

加速度センサ出力値変換テーブル記憶領域34cには、XY軸加速度センサ31およびZ軸接点スイッチ32の出力値を変換してゲームプログラムで利用するための変換テーブルが記憶されている。変換テーブルには、ゲームマップ選択処理用テーブルとプレイヤキャラクタ移動用テーブルとNPC移動用テーブルがある。また、プレイヤキャラクタ移動用テーブルには、空中用、床面用、氷面用および水中用のテーブルがあり、プレイヤキャラクタが存在する座標の地形に応じて選択される。NPC移動用テーブルには表向き用と裏向き用のテーブルがあ

る。N P Cである亀は表向きの状態と裏向きの状態があり、この状態に応じてテーブルが選択される。これらのテーブルの詳細については、図20から図26を参照して後述する。

#### 【0031】

ゲームプログラム記憶領域34eには、C P U 2 1によって実行されるゲームプログラムが記憶される。具体的には、図27を参照して後述するメインプログラム、図28を参照して後述するO G設定プログラム、図29を参照して後述するニュートラルポジション設定プログラム、図30を参照して後述するゲームマップ選択プログラム、図31を参照して後述するセンサ出力読取プログラム、図32から図36を参照して後述するオブジェクト移動プログラム、図37を参照して後述する衝突プログラム、図40を参照して後述する画面スクロールプログラムの他、N P C自律移動プログラムやその他のプログラムが記憶される。

#### 【0032】

図17は、ワークR A M 2 6のメモリマップである。ワークR A M 2 6には、C P U 2 1がゲームプログラムを実行する際の一時的なデータが記憶される。具体的には、ニュートラルポジションデータ記憶領域26a、加速度センサ出力値記憶領域26b、衝撃入力フラグ記憶領域26c、マップ選択画面のカメラ座標記憶領域26e、ゲームマップナンバー記憶領域26fおよびキャラクタデータ記憶領域26gが含まれる。

#### 【0033】

ニュートラルポジションデータ記憶領域26aには、図29を参照して後述するニュートラルポジション設定処理において設定されるニュートラルポジションデータ（N P x, N P y, N P z）が記憶される。これは、ゲームプレイをするときのゲーム装置の基準の傾きに関するデータである。

#### 【0034】

加速度センサ出力値記憶領域26bには、X Y軸加速度センサ31とZ軸接点スイッチ32によって検出され、センサインターフェース33を介して図31のセンサ出力読取処理において読み出される加速度センサ出力値（I N x, I N y, I N z）が記憶される。衝撃入力フラグ記憶領域26cには、X軸方向の加速

度入力をX軸方向のベクトル成分としY軸方向の加速度入力をY軸方向のベクトル成分として合成したベクトルの大きさが一定値以上であったときに1となる衝撃入力フラグ(FS)が記憶される。衝撃入力の判定は、図31のセンサ出力読取処理でおこなわれる。

#### 【0035】

マップ選択画面のカメラ座標記憶領域26eには、図30を参照して後述するゲームマップ選択処理におけるゲームマップ選択用マップのうちLCD12に表示される領域の左上隅の座標(Cx, Cy)が記憶される。ゲームマップナンバー記憶領域26fには、図30を参照して後述するゲーム選択処理において、プレイヤが選択したゲームマップに対応したナンバーデータ(MN)が記憶される。

#### 【0036】

キャラクタデータ記憶領域26gには、キャラクタ(プレイヤキャラクタおよびNPC)毎に、移動加速度データ(Ax, Ay, Az), 移動加速度の変化量データ(dAx, dAy, dAz), 速度データ(Vx, Vy, Vz), 座標データ(X, Y, Z), 前回の座標データ(Px, Py, Pz), 現在位置ステータス(SP)およびポーズナンバー(PN)が記憶される。

#### 【0037】

前回の座標(Px, Py, Pz)は、プレイヤキャラクタやNPCが壁等に衝突した場合に前回の座標に戻すために記憶される。現在位置ステータスデータ(SP)は、プレイヤキャラクタが存在する座標の地形に関するデータであり、このデータに基づいて、加速度センサ出力値変換テーブル(空中, 床面, 氷面, 水面)が選択される。ポーズナンバー(PN)は、キャラクタの状態(ポーズ)に関するデータ(例えば亀の表向きと裏向き等)である。

#### 【0038】

図18は、表示用RAM25のメモリマップである。表示用RAM25には、CPU21がゲームプログラムを実行して得た表示用のデータが一時的に記憶される。表示用RAM25は、オブジェクトデータ記憶領域25a, スクロールカウンタデータ記憶領域25bおよびマップデータ記憶領域25cを備える。オブ

ジェクトデータ記憶領域25aには、ゲームに登場する全キャラクタのうちLCD12の表示領域内に存在するキャラクタについてのデータが記憶される。具体的には、X座標、Y座標、キャラクタIDおよびポーズナンバーが記憶される。

#### 【0039】

スクロールカウンタデータ記憶領域25bには、ゲームマップのうちLCD12に表示される領域の左上隅の座標が記憶される。マップデータ記憶領域25cには、ゲームマップのうちLCD12に表示される領域内のゲームマップデータが記憶される。

#### 【0040】

図19は、バックアップRAM35のメモリマップである。バックアップRAM35には、図28を参照して後述する0G設定処理において設定される0Gポジションデータが記憶される。この0GポジションデータはXY軸加速度センサ31が誤差を有するためにゲーム装置を水平に保ってもセンサ出力値が0にならないことに対処するためのものであり、ゲーム装置を水平に保ったときのセンサ出力値が0GポジションデータとしてバックアップRAM35に記憶され、ゲーム処理において、センサ出力値から差し引かれる。

#### 【0041】

図20から図26は、プログラムROM34の加速度センサ出力値変換テーブル記憶領域34cに記憶される変換テーブルの詳細を示した図である。このテーブルには、XY軸加速度センサ31およびZ軸接点スイッチ32のセンサ出力値(INx, INy, INz)および衝撃入力フラグ(FS)をゲーム処理に利用する際の利用方法や最大値を制限する等の補正処理のためのデータが記憶される。具体的には利用方法、補正比率、特殊補正条件および特殊補正数のデータが記憶される。このテーブルは複数記憶されており、ゲームマップ選択処理用テーブル、プレイヤキャラクタ移動用テーブルおよびNPC移動用テーブルが含まれる。

#### 【0042】

図20のゲームマップ選択処理用テーブルは、図30を参照して後述するゲームマップ選択処理において参照される。このテーブルによって、XY軸加速度セ

ンサの出力値 (INx, INy) はカメラ座標 (Cx, Cy) の変化量の計算をするために利用される。なお、補正比率は2倍なので、XY軸加速度センサ31の出力値 (INx, INy) の2倍だけカメラ座標 (Cx, Cy) は座標移動する。Z軸接点スイッチ出力値 (INz) はマップ決定処理のために利用される。衝撃入力フラグ (FS) は利用しない。

## 【0043】

図21から図24のプレイヤキャラクタ移動用テーブルは、図33を参照して後述するプレイヤキャラクタ移動処理のステップ33における傾き移動処理とステップ34における衝撃移動処理において参照される。プレイヤキャラクタ移動用テーブルは、空中用、床面用、氷面用および水中用のテーブルを含む。プレイヤキャラクタが存在する座標の地形（現在位置ステータス）に応じてこれら複数の変換テーブルのうちいずれか一つが選択されて参照される。

## 【0044】

プレイヤキャラクタ移動用テーブルにおいて、XY軸加速度センサ31の出力値X (INx) は、プレイヤキャラクタのX移動加速度の変化量 (dAx) の計算のために利用され、出力値Y (INy) は、Y移動加速度の変化量 (dAy) の計算のために利用される。現在位置ステータスが「空中」の場合は、図21を参照して移動加速度の変化量 (dAx, dAy) はゼロである。「床面」の場合は、図22を参照すると補正比率が2倍であるから、XY軸加速度センサ31の出力値 (INx, INy) の2倍が移動加速度の変化量 (dAx, dAy) となる。また、特殊補正条件1によりXY軸加速度センサ31の出力値 (INx, INy) が20より大きい場合には移動加速度の変化量 (dAx, dAy) は40に制限される。「氷面」の場合は、図23を参照してXY軸加速度センサ31の出力値 (INx, INy) の3倍が移動加速度の変化量 (dAx, dAy) となる（「氷面」では移動量大きい）。また、特殊補正条件1によりXY軸加速度センサ31の出力値 (INx, INy) が20より大きい場合には移動加速度の変化量 (dAx, dAy) は60に制限される。「水中」の場合は、図24を参照してXY軸加速度センサ31の出力値 (INx, INy) の1/2倍が移動加速度の変化量 (dAx, dAy) となる（「水中」では移動量小さい）。また、特

殊補正条件1によりXY軸加速度センサ31の出力値(INx, INy)が10より大きい場合には移動加速度の変化量(dAx, dAy)は5に制限される。

## 【0045】

プレイヤキャラクタ移動用テーブルにおいて、Z軸接点スイッチ32の出力値(INz)は、プレイヤキャラクタのZ移動加速度の変化量(dAz)の計算のために利用される。特殊補正条件はない。

## 【0046】

プレイヤキャラクタ移動用テーブルにおいて、衝撃入力フラグ(FS)は、X移動加速度とY移動加速度の変化量(dAx, dAy)に影響を与える。現在位置ステータスが「空中」および「水中」の場合は図21および図24を参照して衝撃入力フラグ(FS)は無視される。現在位置ステータスが「床面」の場合は、図22を参照して、X移動加速度およびY移動加速度の変化量(dAx, dAy)をそれぞれ3倍する処理がされる。現在位置ステータスが「水面」の場合は、図23を参照して、X移動加速度およびY移動加速度の変化量(dAx, dAy)をそれぞれ5倍する処理がされる。このようにして、衝撃入力があった場合、「床面」と「水面」ではX移動加速度およびY移動加速度の変化量(dAx, dAy)が通常に比べて増加される(高速移動する)。

## 【0047】

図25および図26のNPC移動用テーブルは、図34を参照して後述するNPC移動処理のステップ44における傾き移動処理とステップ45における衝撃移動処理において参照される。NPC移動用テーブルは、表向き用と裏向き用のテーブルを含む。NPCである亀のポーズ(表向きか裏向きか)に応じてこれら2つの変換テーブルのうちいずれか一つが選択されて参照される。

## 【0048】

NPC移動用テーブルにおいて、XY軸加速度センサ31の出力値X(INx)は、NPCのX移動加速度の変化量(dAx)の計算のために利用され、出力値Y(INy)は、Y移動加速度の変化量(dAy)の計算のために利用される。「表向き」の場合は、図25を参照すると補正比率が1/2倍であるから、XY軸加速度センサ31の出力値(INx, INy)の1/2倍がX移動加速度お

およびY移動加速度の変化量（d Ax, d Ay）となる。また、特殊補正条件1によりXY軸加速度センサ31の出力値（INx, INy）が10より小さい場合には移動加速度の変化量（d Ax, d Ay）は0になる（「表向き」の場合は少しの傾き入力では亀は踏ん張るので滑らない）。また、特殊補正条件2によりXY軸加速度センサ31の出力値（INx, INy）が20より大きい場合には、移動加速度の変化量（d Ax, d Ay）は10に制限される。「裏向き」の場合は、図26を参照してXY軸加速度センサ31の出力値（INx, INy）の2倍がX移動加速度およびY移動加速度の変化量（d Ax, d Ay）となる（「表向き」に比べて「裏向き」の方が亀は滑りやすいので移動量が大きい）。また、特殊補正条件1によりXY軸加速度センサ31の出力値（INx, INy）が20より大きい場合には、移動加速度の変化量（d Ax, d Ay）は40に制限される。

## 【0049】

NPC移動用テーブルにおいて、Z軸接点スイッチの出力値（INz）は亀の表裏の逆転判定に利用される。Z軸接点スイッチ出力値が1になる毎に亀は表裏の状態を繰り返す。衝撃入力フラグ（FS）はNPC移動処理には利用しない。

## 【0050】

図27はメインルーチンのフローチャートである。ゲーム装置本体10にカートリッジ30を差して、ゲーム装置本体10の電源をONにすると、図27に示すメインルーチンが開始される。まずステップ11において、初回起動時であるかまたはプレイヤがOG設定リクエスト（例えば、図1の操作スイッチ13bを押しながら起動）があったか否かが判断される。初回起動時でなくOG設定リクエストもない場合はステップ13に進む。初回起動時かまたはOG設定リクエストがあった場合は、ステップ12において、図28を参照して後述するOG設定処理が行われた後、ステップ14に進む。ステップ14において、図29を参照して後述するニュートラルポジション設定処理が行われた後、ステップ17に進む。ここで、ニュートラルポジション設定とは、ゲームプレイする際のゲーム装置の基準の傾きを設定することであり、お勧めポジション設定とは、ゲーム内容に応じて適切なニュートラルポジションに関するデータ（プログラムROM34

のお勧めポジション照準目標座標34d)を予めゲームプログラムに記憶させておいて、そのデータに基づいてニュートラルポジションを設定することである。

#### 【0051】

ステップ17において、図30を参照して後述するゲームマップ選択処理が行われ、複数のゲームマップのうちのいずれか一つがプレイヤによって選択される。ステップ17の後、メインループに進む。

#### 【0052】

ステップ19からステップ29までがメインループであり、ゲームオーバーになるかまたはゲームクリアになるまで繰り返し処理される。ステップ19において、ワークRAM26のキャラクタデータ26gの座標(X, Y, Z)およびポーズナンバー(PN)、プログラムROM34のオブジェクトキャラクタデータ34aおよびマップデータ34bに基づいて表示用RAM25に必要なデータが書き込まれ、表示用RAM25に記憶されたデータに基づいてLCD12にゲーム画面が表示される。ステップ20において、図31を参照して後述するセンサ出力読取処理が行われて、XY軸加速度センサ31およびZ軸接点スイッチ32の出力値がセンサインターフェース33を介して読取られ、補正される。ステップ20の後、ステップ21においてニュートラルポジション設定要求があったか否かが判断され、要求がない場合はステップ23に進み、要求があった場合はステップ22に進んでニュートラルポジション設定処理が行われ、ニュートラルポジションの再設定がされた後、ステップ19に戻る。これは、一つの操作スイッチ(例えば図1における操作スイッチ13e)をニュートラルポジション設定専用の操作スイッチに割り当てておいて、この操作スイッチ13eを押すことによってゲーム中においても、常にニュートラルポジションの再設定を可能にすることを意味する。

#### 【0053】

ステップ23において、衝撃入力フラグがONであるか否かが判断される。衝撃入力フラグがOFFのばあいはステップ26に進む。衝撃入力フラグがONの場合はステップ24に進み、プレイヤキャラクタの現在座標の地形が水中であるか否かが判断される(現在位置ステータス(PS)に基づいて判断される)。水

中でないと判断されると、ステップ26に進む。水中であると判断されるとステップ25において波の発生処理が行われる（前述の15中段に示すような画面表示となる）。具体的には、センサ出力値X（INx）をX軸方向のベクトル成分としセンサ出力値Y（INY）をY軸方向のベクトル成分として合成したベクトルの方向に、合成ベクトルの大きさに応じた波を発生する処理をする。プレイヤはあたかも自分がゲーム装置に与えた衝撃がそのままゲーム空間中の環境（水）に反映されたかのような感覚をプレイヤに与えることができる。ステップ25の後、ステップ26に進む。

#### 【0054】

ステップ26において、図32から図36を参照して後述する各オブジェクト移動処理が行われ、プレイヤキャラクタおよびNPCの移動処理が行われる。ステップ26の後、ステップ27において、図37を参照して後述する衝突処理が行われ、プレイヤキャラクタとNPC等との衝突処理が行われる。ステップ27の後、ステップ29において図40を参照して後述する画面スクロール処理が行われる。

#### 【0055】

図28は0G設定処理のフローチャートである。このサブルーチンでは、ゲーム装置（具体的にはLCD12の表示面）を水平に把持したときのXY軸加速度センサ31の出力値を0GポジションデータとしてバックアップRAM35に記憶しておく処理を行う。

#### 【0056】

まずステップ121において、「地面と水平にあわせたら操作スイッチを押してください」という表示がLCD12にされて、ゲーム装置（具体的にはLCD12の表示面）が水平になるようにプレイヤに要求する。ステップ122において操作スイッチの入力処理が行われ、ステップ123において、決定のための操作スイッチ（例えば図1における操作スイッチ13b）が押されたことが判断されると、ステップ124においてZ軸接点スイッチがONか否かが判断される。Z軸接点スイッチがONの場合は、ステップ125において警告音を発生してステップ121に戻る。Z軸接点スイッチがONになっている場合は、LCD12

の表示面が下方を向いた状態になっているので、プレイヤに再度設定することを要求するものである。ステップ124において、Z軸接点スイッチがOFFであると判断された場合は、ステップ126において、このときのXY軸加速度センサ31の出力値がバックアップRAM35に0Gポジションデータとして記憶される。

#### 【0057】

図29は、ニュートラルポジション設定処理のフローチャートである。このサブルーチンでは、ゲームプレイしやすいゲーム装置の把持角度をプレイヤが任意に決定して、そのときのXY軸加速度センサ31およびZ軸接点スイッチ32の出力値をニュートラルポジションデータとしてワークRAM26に記憶しておく処理を行う。

#### 【0058】

まずステップ141において、「遊びやすい角度にあわせたら操作スイッチを押してください」という表示がLCD12にされる。ステップ142において、操作スイッチ入力処理が行われ、ステップ143において、決定のための操作スイッチ（例えば図1の操作スイッチ13b）が押されたことが判断されると、ステップ144において、このときのXY軸加速度センサ31の出力値から前述の0Gポジションデータを引く補正がされた後（ニュートラルポジションデータは水平状態からの傾きに対応したデータとする）、ステップ145において、XY軸加速度センサの出力補正值（ステップ144の計算結果）とZ軸接点スイッチ32の出力値がワークRAM26のニュートラルポジションデータ記憶領域26aにニュートラルポジションデータとして記憶される。

#### 【0059】

図30はゲームマップ選択処理のフローチャートである。このサブルーチンでは、ゲームプログラムに記憶されている複数のゲームマップのうちからプレイヤーがいずれか一つを選択する。ゲームマップ選択処理の画面は例えば前述の図13のように表示される。LCD12にはゲームマップ選択用マップの一部領域が表示される。プレイヤは、X軸方向またはY軸方向にスライド入力することによって表示領域を移動させて、マップアイコン（図13のA, B, C, D）を表示領

域内に表示されるようにしてからZ軸方向に運動入力する。Z軸方向に運動入力（または衝撃入力）したときに表示領域内に表示されているコースアイコンに対応したゲームコースが選択されることになる。

#### 【0060】

まずステップ171において、カメラ座標（Cx, Cy）が初期化される。その後、ステップ172において、カメラ座標（Cx, Cy）に基づいてゲームマップ選択用マップの一部領域がLCD12に表示される。ステップ173において、図31を参照して後述するセンサ出力読取処理が行われて、XY軸加速度センサ31およびZ軸接点スイッチ32の出力値が読取られ、補正される。ステップ174において、図20に示すゲームマップ選択処理用テーブルが参照されて、センサ出力値（INx, INy）に基づいてカメラ座標（Cx, Cy）が変化される。具体的には、補正比率が2倍であるから、センサー出力値（INx, INy）の2倍の値だけカメラ座標（Cx, Cy）を変化させる。例えばセンサー出力値（INx）の値が5であるときは、カメラ座標（Cx）が+10される。ステップ175において、カメラ座標（Cx, Cy）に基づく表示領域がゲームマップ選択用マップの範囲外であるか否かが判断され、範囲外でない場合はステップ177に進み、範囲外である場合は、ステップ176において、ゲームマップ選択用マップの端領域が表示されるように補正された後、ステップ177に進む。ステップ177において、Z軸接点スイッチ32がONであるか否かが判断される。Z軸接点スイッチ32がOFFであると判断されると、ステップ172に戻る。Z軸接点スイッチ32がONであると判断されると、ステップ178において、マップアイコン（図13のA, B, C, D）のうちのいずれか一つがLCD12の表示領域内に表示されているか否かが判断される。マップアイコンが表示領域内に表示されていないことが判断されると、ステップ179において、警告音を発生してステップ172に戻る。マップアイコンが表示領域内に表示されていることが判断されると、ステップ181において、表示されているマップアイコンに対応したゲームマップナンバー（MN）がワークRAM26に記憶される。

#### 【0061】

図31はセンサ出力読取処理のフローチャートである。このサブルーチンでは、XY軸加速度センサ31およびZ軸接点スイッチ32の出力値が読取られ、補正される。具体的には、センサインターフェース33のラッチ334およびラッチ335のデータから加速度センサ出力値(INx, INy)およびZ軸接点スイッチ出力値(INz)が読み出される。さらに、0Gポジションデータおよびニュートラルポジションデータで補正する処理がされる。

#### 【0062】

ステップ201において、センサインターフェース33のラッチ334およびラッチ335のデータが読み込まれる。ステップ202において、ラッチデータから加速度センサ出力値(INx, INy)およびZ軸接点スイッチ出力値(INz)が読み出されてワークRAM26の加速度センサ出力値記憶領域26bに記憶される。ステップ203において、衝撃入力があったか否かが判断される。具体的には、加速度センサ出力値X(INx)をX軸方向のベクトル成分とし、加速度センサ出力値Y(INy)をY軸方向のベクトル成分として合成したベクトルの大きさが一定値以上か否かが判断される。一定値以上であると判断されると、ステップ204において、衝撃入力フラグ(FS)がONに設定された後、ステップ206に進む。合成ベクトルの大きさが一定値より小さいと判断されると、ステップ205において、衝撃入力フラグ(FS)がOFFに設定された後、ステップ206に進む。ステップ206において、ステップ202において記憶された加速度センサ出力値記憶領域26bのデータからバックアップRAM35に記憶された0Gポジションデータを引く処理がされる。ステップ206の後、ステップ207において、さらにニュートラルポジションデータで補正した値が、加速度センサ出力記憶領域26bにINx, INyおよびINzとして再記憶される。

#### 【0063】

ニュートラルポジションデータによる補正は、具体的には、加速度センサ出力値X(INx)および加速度センサ出力値Y(INy)については、ニュートラルポジションデータ(NPx, NPy)の値を引く処理がされる。Z軸接点スイッチ出力値(INz)については、ニュートラルポジションデータ(NPz)の

値が1の場合に、0と1を反転させる処理がされる。

#### 【0064】

図32から図36はオブジェクト移動処理のフローチャートである。図32は、オブジェクト移動処理のメインルーチンのフローチャートである。ステップ261において、図33を参照して後述するプレイヤキャラクタ移動処理が行われる。ステップ262において、図34を参照して後述するNPC移動処理が行われる。このNPC移動処理はNPCの数だけ繰り返される。

#### 【0065】

図33は、プレイヤキャラクタ移動処理のフローチャートである。ステップ31において、プレイヤキャラクタの現在の座標(X, Y, Z)が、前回の座標(Px, Py, Pz)としてコピーされて記憶される。これは、図37を参照して後述する衝突処理において、プレイヤキャラクタが壁に衝突したときに前回の座標に戻すために必要である。ステップ32において、移動加速度の変化量(dAx, dAy, dAz)が初期化された後、ステップ33において、傾き移動処理が行われる。傾き移動処理では、前述の図21から図24に示す変換テーブルのうちプレイヤキャラクタの現在位置ステータスに応じて適切なものを参照して、プレイヤキャラクタのX移動加速度とY移動加速度の変化量を算出する処理が行われる。この処理によってゲーム装置の傾き(傾き入力)に応じてキャラクタが転がる(滑る)ように移動加速度の変化量(dAx, dAy)が決定される。さらに、ステップ34において、衝撃移動処理が行われる。衝撃移動処理では、前述の図21から図24に示す変換テーブルのうち適切なものを参照して、プレイヤキャラクタのX移動加速度とY移動加速度の変化量を増加させる処理が行われる。この処理によって、衝撃入力がされたときに、プレイヤキャラクタがダッシュ(高速移動)するように移動加速度の変化量(dAx, dAy)を増加する処理がされる。ステップ35において、図35を参照して後述するジャンプ移動処理が行われる。ステップ35の後、ステップ36において、前述の図27のフローチャートにおけるステップ25において波発生処理がされたか否かが判断される。波が発生していないことが判断されると、ステップ38に進む。波が発生したことが判断されると、ステップ37において、図36を参照して後述する波移

動処理が行われた後、ステップ38に進む。ステップ38において、ステップ33からステップ37までの傾き移動処理、衝撃移動処理、ジャンプ処理、波移動処理で計算された移動加速度の変化量（dAx, dAy, dAz）に基づいて移動加速度（Ax, Ay, Az）が計算され、移動加速度（Ax, Ay, Az）に基づいて速度（Vx, Vy, Vz）が計算される。ステップ39において、速度（Vx, Vy, Vz）に基づいて座標（X, Y, Z）が計算される。

#### 【0066】

図34はNPC移動処理のフローチャートである。ステップ41において、現在の座標（X, Y, Z）が前回の座標（Px, Py, Pz）にコピーして記憶される。ステップ42において、移動加速度の変化量（dAx, dAy, dAz）が初期化される。ステップ43において、ゲームプログラムに基づくNPCの自律移動処理が行われる。具体的には例えば亀は乱数値に基づいて移動加速度の変化量（dAx, dAy, dAz）が決定される。ステップ43の後、ステップ44において、傾き移動処理が行われる。傾き移動処理では、前述の図25または図26に示す変換テーブルのうちNPCのポーズナンバーに応じて適切なものを参照して、NPCのX移動加速度とY移動加速度の変化量を算出する処理が行われる。さらに、ステップ45において、衝撃移動処理が行われるが、本実施例の場合は、NPCは衝撃入力による影響を受けない。ステップ45の後、ステップ46において、前述の図27のフローチャートにおけるステップ25において波発生処理がされたか否かが判断される。波が発生していないことが判断されると、ステップ48に進む。波が発生したことが判断されると、ステップ47において、図36を参照して後述する波移動処理が行われた後、ステップ48に進む。ステップ48において、ステップ43からステップ47までの自律移動処理、傾き移動処理、衝撃移動処理、波移動処理で計算された移動加速度の変化量（dAx, dAy, dAz）に基づいて移動加速度（Ax, Ay, Az）が計算され、移動加速度（Ax, Ay, Az）に基づいて速度（Vx, Vy, Vz）が計算される。ステップ49において、速度（Vx, Vy, Vz）に基づいて座標（X, Y, Z）が計算される。ステップ51において、Z軸接点スイッチ出力値（INz）が1か否かが判断される。Z軸接点スイッチ出力値（INz）が0の場合はN

PC移動処理サブルーチンを終了する。Z軸接点スイッチ出力値（INz）が1の場合は、ステップ52において表向きと裏向きの反転処理をする。具体的には、ワークRAM26のキャラクタデータのポーズナンバー（PN）を変化させる。

#### 【0067】

図35はジャンプ処理のフローチャートである。このサブルーチンでは、Z軸方向の運動入力があったとき、プレイヤキャラクタをジャンプさせる処理がされ、Z軸方向の運動入力がなくてかつプレイヤキャラクタが「空中」にいるときに降下させる処理がされる。

#### 【0068】

ステップ351において、Z軸接点スイッチ出力値（INz）が1か否かが判断される。Z軸接点スイッチ出力値（INz）が1の場合には、ステップ352において、現在位置ステータス（PS）が「空中」に設定された後、ステップ353において、Z移動加速度の変化量（dAz）が1にされる。ステップ351において、Z軸接点スイッチ出力値（INz）が0であると判断されると、ステップ354において、プレイヤキャラクタが「空中」にいるか否かが判断され、「空中」にいない場合はジャンプ処理を終了する。ステップ354において、「空中」にいる場合は、ステップ355において、Z移動加速度の変化量（dAz）が-1にされた後、ジャンプ処理を終了する。

#### 【0069】

図36は波移動処理のフローチャートである。このサブルーチンでは、プレイヤの衝撃入力により発生した波によるプレイヤキャラクタやNPCの移動加速度の変化量を計算する処理がされる。ステップ361において、現在位置ステータスが読み込まれ、ステップ362において、波の影響を受ける位置か否か（すなわち「水中」か否か）が判断される。波の影響を受けない位置であると判断されると、波移動処理を終了する。波の影響を受ける位置であると判断されると、ステップ363において、波の影響によるX移動加速度の変化量とY移動加速度の変化量が計算されて、傾き移動処理および衝撃移動処理において計算されたX移動加速度の変化量とY移動加速度の変化量に加算される。

## 【0070】

図37は衝突処理のフローチャートである。ステップ271からステップ275においてNPC衝突判定処理が行われる。このNPC衝突判定処理はNPCの数だけ繰り返される。ステップ271において、NPCが壁と衝突したか否かが判断される。衝突したと判断されるとステップ273に進む。壁と衝突していないと判断されるとステップ272に進み、別のNPCと衝突したか否かが判断される。別のNPCと衝突したと判断されるとステップ273に進む。別のNPCと衝突していないと判断されるとステップ275に進む。壁または別のNPCに衝突したと判定された場合は、ステップ273において、衝突音が発生された後、ステップ274において、NPCの座標(X, Y, Z)を前回の座標(Px, Py, Pz)に戻す処理が行われた後、ステップ275に進む。

## 【0071】

ステップ275において、NPCの現在位置ステータスが検出されてワークRAM26に記憶される。ステップ275の後、ステップ276において、プレイヤキャラクタが壁と衝突したか否かが判断される。壁に衝突していないと判断された場合には、ステップ279に進む。壁に衝突したと判断された場合には、ステップ277において、衝突音が発生された後、ステップ278において、プレイヤキャラクタの座標(X, Y, Z)を前回の座標(Px, Py, Pz)に戻す処理がされた後、ステップ279に進む。

ステップ279において、プレイヤキャラクタの現在位置ステータスが検出されてワークRAM26に記憶される。ステップ279の後、ステップ281において、プレイヤキャラクタがNPCと衝突したか否かが判断される。NPCと衝突したと判断されると、ステップ282において、NPCを消滅する処理がされる。ステップ282の後、ステップ283において、すべてのNPCが消滅したか否かが判断される。すべてのNPCが消滅したことが判断されるとステップ284においてゲームクリア処理が行われる。ステップ281において、NPCと衝突していないと判断された場合およびステップ283においてすべてのNPCが消滅していないと判断された場合にはステップ285に進む。ステップ285において、プレイヤキャラクタが穴に落ちたか否かが判断される。穴に落ちたと判

断された場合は、ステップ286においてゲームオーバー処理が行われる。穴に落ちていないと判断された場合は、衝突処理を終了する。

## 【0072】

図38および図39は画面スクロールを示す画面の一例である。画面にはプレイヤーキャラクタであるボール61、NPCである亀62a～62c、迷路を構成する壁63および穴64が表示されている。点線65は画面スクロールの限界を示すものである（点線65は実際にはLCD12には表示されない）。前述のようにゲームマップはLCD12の表示領域より大きな仮想マップであり、LCD12にはゲームマップのうちプレイヤーキャラクタ61の周辺の一部領域が表示される。プレイヤーがゲーム装置を傾ける等してプレイヤーキャラクタ61が点線65より外側領域に移動しようとすると、画面をスクロールしてLCD12に表示されるゲームマップ表示領域を移動させ、さらにプレイヤーキャラクタ61およびNPC62をスクロールした分だけ画面の中央方向に移動表示する。この画面スクロールによって、より広いゲームマップ上でのゲームを楽しむことができる。

## 【0073】

例えば、図38に示すようにプレイヤーキャラクタが点線65を超えて左側の領域に移動しようとすると、ゲームマップの表示領域を左にスクロールさせ、プレイヤーキャラクタ61およびNPC62をスクロールさせた分だけ右に移動表示させる（図39）。なお、傾き入力の大きさに応じてスクロールする速度を変化させても良い。

## 【0074】

図40は、画面スクロール処理のフローチャートである。ステップ291において、プレイヤーキャラクタがスクロールエリアをX軸のマイナス方向に外れたか否かが判断される。ここで、スクロールエリアとは、図38における点線65で囲まれる領域のことである。X軸のマイナス方向に外れていないことが判断されるとステップ294に進む。X軸のマイナス方向に外れたことが判断されると、ステップ292において、LCD12に現在表示されている領域がゲームマップの左端領域か否かが判断される。左端領域であると判断された場合は、ステップ294に進む。左端領域でないと判断された場合は、ステップ293において、表

示用RAM25に記憶されたスクロールカウンタX座標(SCx)を一定量減らす処理がされた後、ステップ294に進む。ステップ294において、プレイヤキャラクタがスクロールエリアをX軸のプラス方向に外れたか否かが判断される。X軸のプラス方向に外れていないことが判断されると、ステップ297に進む。X軸のプラス方向に外れたことが判断されると、ステップ295において、LCD12に現在表示されている領域がゲームマップの右端領域か否かが判断される。右端領域であることが判断されるとステップ297に進む。右端領域でないと判断された場合は、ステップ296において、スクロールカウンタX座標(SCx)を一定量増やす処理がされた後、ステップ297に進む。

#### 【0075】

ステップ297において、プレイヤキャラクタがスクロールエリアをY軸のマイナス方向に外れたか否かが判断される。Y軸のマイナス方向に外れていないことが判断されると、ステップ301に進む。Y軸のマイナス方向に外れたことが判断されると、ステップ298において、LCD12に現在表示されている領域がゲームマップの上端領域か否かが判断されるとステップ301に進む。上端領域でないと判断された場合は、ステップ299において、スクロールカウンタY座標(SCy)を一定量減らす処理がされた後、ステップ301に進む。ステップ301において、プレイヤキャラクタがスクロールエリアをY軸のプラス方向に外れたか否かが判断される。Y軸のプラス方向に外れていないことが判断されると、画面スクロール処理を終了する。Y軸のプラス方向に外れたことが判断されると、ステップ302において、LCD12に現在表示されている領域がゲームマップの下端領域か否かが判断される。下端領域であることが判断されると、画面スクロール処理を終了する。下端領域でないと判断された場合は、ステップ303において、スクロールカウンタY座標(SCy)を一定量増やす処理がされた後、画面スクロール処理を終了する。

#### 【0076】

##### (第2実施例)

次に、図41～図49を参照して、本発明の第2の実施例のゲーム装置を説明する。第2の実施例のゲーム装置の外観図、XYZ軸定義図、ブロック図、セン

サインインターフェースの計測原理図、Z軸接点スイッチの構造図は、第1の実施例における図1～図7と共通であり、説明を省略する。

#### 【0077】

図41は、本実施例のゲーム画面の一例を示した図である。このゲームは、プレイヤーがゲーム装置に衝撃を与えることによってゲーム空間の地形を隆起させることにより、ゲームキャラクタの移動を制御して楽しむゲームである。

#### 【0078】

図41(a)に示すように、ゲームキャラクタである亀81と地形隆起キャラクタ82がゲーム画面に表示される。図41(b)に示すように、亀81はゲームプログラムによって自立的に移動する。図41(c)に示す状態において、ゲーム装置にZ軸方向の衝撃入力を与えると、図41(c)に示すように、地形隆起キャラクタ82は隆起して高く大きく表示され、それによって亀81は滑る移動制御をされる(前進していた亀82が地形隆起によって後退する)。このような処理をすることによって、ゲーム装置にZ軸方向の衝撃を与えたときにゲーム空間である地形がエネルギーを与えられて地形が隆起するかのような感覚をプレイヤーに与えることができる。

#### 【0079】

図42はZ軸方向の衝撃入力による地形隆起処理を示すゲーム画面の一例である。図42(a)において、外枠12'はゲーム空間全体を示しており、内枠12はLCD12に表示される表示領域を示している。ゲーム空間はLCD12の表示領域より大きな世界であり、LCD12にはゲーム空間の一部領域が表示される。ゲーム空間には地形隆起キャラクタ82が12個存在し(82a～82l)、亀キャラクタ81が3匹存在する(81a～81c)。このうち、LCD12には4個の地形隆起キャラクタ(82a, 82b, 82e, 82f)と1匹の亀キャラクタ(81a)が表示されている。

#### 【0080】

図42(a)に示す状態において、ゲーム装置にZ軸方向の衝撃入力を与えると、図42(b)に示すように、12個の地形隆起キャラクタ(82a～82l、ゲーム空間全体の地形隆起キャラクタ)は1段階隆起して高く大きく表示され

る。このとき、地形が隆起した地点に存在する亀キャラクタ（81aおよび81b）は地形の隆起によって滑って移動する表示がされる。

#### 【0081】

図42(b)に示す状態において、Aボタン（操作スイッチ13b）を操作しつつZ軸方向の衝撃入力を与えると、LCD12に表示されている4個の地形隆起キャラクタ（82a, 82b, 82e, 82f）のみさらに1段階隆起して高く大きく表示される。このときも同様に、地形が隆起した地点に存在する亀キャラクタ（81a）は地形の隆起によって滑って移動する表示がされる。このように処理することによって、Aボタンを押しつつZ軸方向の衝撃入力を与えた場合には、LCD12に表示された領域に限定したゲーム空間に衝撃によるエネルギーが与えられたかのような感覚をプレイヤに与えることができる。

#### 【0082】

なお、図示しないが、図42(b)に示す状態において、Bボタン（操作スイッチ13c）を操作しつつZ軸方向の衝撃入力を与えると、LCD12に表示されていない8個の地形隆起キャラクタ（82c, 82d, 82g, 82h, 82i～82l）のみ1段階隆起して高く大きく表示される。このときも同様に、地形が隆起した地点に存在する亀キャラクタ（81b, 81c）は地形の隆起によって滑って移動する表示がされる。このように処理することによって、Bボタンを押しつつZ軸方向の衝撃入力を与えた場合には、LCD12に表示されていない領域に限定してゲーム空間に衝撃によるエネルギーが与えられたかのような感覚をプレイヤに与えることができる。

#### 【0083】

図43は、ゲーム空間の表示領域のスクロール処理を示すゲーム画面の一例である。ゲーム装置をスライド入力（第1の実施例における図9を参照）をすることによってゲーム空間の表示領域がスクロールする。例えば、図43(a)では、LCD12に地形隆起キャラクタ82a, 82b, 82e, 82fおよび亀キャラクタ81aが表示されている。この状態において、ゲーム装置をY軸のマイナス方向にスライドさせると、ゲーム空間の表示領域は下方にスクロールして、図43(b)に示すように、地形キャラクタ82e, 82fおよび亀キャラクタ

81aが表示されることになる。

#### 【0084】

また、図43(b)に示す状態において、ゲーム装置をX軸のプラス方向にスライドさせると、ゲーム空間の表示領域は右方にスクロールして、図43(c)に示すように、地形キャラクタ82fおよび亀キャラクタ81aが表示されることになる。このような処理をすることによって、プレイヤーはLCD12より大きなゲーム空間でゲームを楽しむことができる。また、前述のように、AボタンやBボタンによって表示領域の内外に限定してゲーム空間に影響を与える（地形を隆起させる）ことができるので、複雑なゲームを楽しむことができる。

#### 【0085】

図44は、XY軸方向の衝撃入力による温度上昇画面制御を示す図である。亀キャラクタ81a～81cは、前述のようにゲームプログラムによって自立的な移動をするが、この自律移動は、温度が上昇することによって活発になる（具体的には移動量が増加する）。図44(a)に示す状態において、XY軸方向の衝撃入力（第1の実施例における図11を参照）をすると、温度パラメータが上昇し、亀キャラクタ81a～81cが活発に移動する表示がされる。このような処理をすることによって、ゲーム装置にXY軸方向の衝撃を与えたときにゲーム空間にエネルギーが与えられて温度が上昇するかのような感覚をプレイヤーに与えることができる。

#### 【0086】

以下、図45および図46を参照して、メモリに記憶されるデータを説明する。

図45は、プログラムROM34のメモリマップである。プログラムROM34には、CPU21によって実行されるゲームプログラムおよびゲームデータが記憶される。プログラムROM34は、具体的には、オブジェクトキャラクタデータ記憶領域342a、マップデータ記憶領域342b、地形隆起ポイントデータ記憶領域342c、スクロールの限界値データ記憶領域342d、加速度センサ出力値変換テーブル記憶領域342eおよびゲームプログラム記憶領域342fを含む。オブジェクトキャラクタデータ記憶領域342aおよびマップデータ

記憶領域342bには、オブジェクトキャラクタおよびゲームマップのグラフィックデータが記憶される。地形隆起ポイントデータ記憶領域342cには、前述の図42に示されるような地形隆起キャラクタ(82a～821)のそれぞれについてゲーム空間における位置データ(X座標およびY座標; Px1～Px12, Py1～Py12)が記憶される。スクロールの限界値データ記憶領域342dには、ゲーム空間をスクロール表示する際に、ゲーム空間の上下左右端になった場合にスクロールをしないようにするためにスクロールの限界値を示すデータ(SCxmax, SCymax)が記憶される。

## 【0087】

加速度センサ出力値変換テーブル記憶領域342dには、XY軸加速度センサ31およびZ軸接点スイッチ32の出力値を変換してゲームプログラムで利用するための変換テーブルが記憶されている。具体的には、前述の第1実施例の変換テーブル(図20から図26)と同様のデータが記憶されており、センサ出力値X(INx)およびセンサ出力値Y(INy)が、後述の図48を参照して後述する視界移動処理において、スクロールカウンタX座標(SCx)およびY座標(SCy)の変化量の計算に利用されることが定義されている。これによってゲーム装置をスライド入力(第1の実施例における図9を参照)をすることによってゲーム空間の表示領域がスクロールし、視界が移動する処理がされることになる。また、Z軸接点スイッチ出力値(INz)は地形の隆起判定に利用されることが定義され、衝撃入力フラグ(FS)は温度上昇の判定に利用されることが定義されている。

## 【0088】

ゲームプログラム記憶領域342fには、CPU21によって実行されるゲームプログラムが記憶される。具体的には、図47を参照して後述するメインプログラム、第1の実施例における図31と同様のセンサ出力読取プログラム、図48を参照して後述する視界移動プログラム、図49を参照して後述する地形隆起プログラム、温度上昇プログラム、亀キャラクタ制御プログラムやその他のプログラムが記憶される。

## 【0089】

図46はワークRAM26のメモリマップである。ワークRAM26には、CPU21がゲームプログラムを実行する際の一時的なデータが記憶される。具体的には、加速度センサ出力値記憶領域262a、衝撃入力フラグ記憶領域262b、地形隆起データ記憶領域262c、温度データ記憶領域262dおよびキャラクタデータ記憶領域262eが含まれる。

## 【0090】

加速度センサ出力値記憶領域262aおよび衝撃入力フラグ記憶領域262bに記憶されるデータは第1の実施例と同様であるので説明を省略する。地形隆起データ記憶領域262cには、それぞれの地形隆起ポイントについての高さデータが記憶される。高さデータは、図49を参照して後述する地形隆起処理においてZ軸方向の衝撃入力に応じて変更される。このデータに基づいてそれぞれの地形隆起ポイントについて地形隆起キャラクタの表示状態が決定される。例えば、高さデータが1の場合には図42(a)の82aのように表示され、高さデータが2の場合には図42(b)の82aのように表示され、高さデータが3の場合には図42(c)の82aのように地形隆起キャラクタが表示される。

## 【0091】

温度データ記憶領域には、ゲーム空間の温度データが記憶される。温度データは、温度上昇処理(図47に示すメインプログラムのステップ64)においてX-Y軸方向の衝撃入力に応じて変更される。このデータは亀キャラクタ制御処理(自律移動、図47に示すメインプログラムのステップ65)に影響を与える。

キャラクタデータ記憶領域262eには、亀キャラクタの数だけ、座標データ(X、Y、Z)および前回の座標データ(Px、Py、Pz)が記憶される。

表示用RAMのメモリマップは第1の実施例における図18と同様であるので説明を省略する。

## 【0092】

以下、図47～図49を参照して、ゲームプログラムの処理の流れを説明する

図47はメインルーチンのフローチャートである。ゲーム装置本体10にカ一

トリッジ30を差して、ゲーム装置本体10の電源をONにすると、図47に示すメインルーチンが開始される。第2の実施例においても、第1の実施例と同じように、OGポジション設定処理やニュートラルポジションの設定処理をおこなっても良いが、説明を簡単にするため省略することにする。

## 【0093】

まずステップ61において、第1の実施例における図31と同様のセンサ出力読取処理が行われて、XY軸加速度センサ31およびZ軸接点スイッチ32の出力値がセンサインターフェース33を介して読取られる（OGポジションデータおよびニュートラルポジションデータによる補正は省略）。ステップ61の後、ステップ62において、図48を参照して後述する視界移動処理（ゲーム空間の表示領域のスクロール処理）が行われる。ステップ62の後、ステップ63において、図49を参照して後述する地形隆起処理が行われる。ステップ63の後、ステップ64において、温度上昇処理が行われる。温度上昇処理では、まずXY軸方向の衝撃入力があったか否かが判定され、XY軸方向の衝撃入力があった場合には、温度パラメータ（T）を1増加する処理がされる。ステップ64の後、ステップ65において、亀キャラクタ制御処理が行われる。亀キャラクタ制御処理では、まず自律移動による亀キャラクタの移動処理が行われる。具体的には、例えば、乱数値によって亀キャラクタの移動量を計算する処理が行われる。なお、温度（T）が高いときには亀キャラクタの自立移動量が大きくなるように制御される。この後、地形隆起による亀キャラクタの移動処理が行われる。具体的には、亀キャラクタの下の地形が隆起した場合に、亀キャラクタを滑らせて移動させる処理がされる。なお、亀キャラクタ制御処理は亀キャラクタの数だけ繰り返される。

## 【0094】

ステップ65の後、ステップ66において、前述の視界移動処理、地形隆起処理、亀キャラクタ制御処理の処理結果に基づいて、ゲーム空間のスクロール表示および地形隆起オブジェクトや亀キャラクタの表示処理がされる。なお、地形隆起処理によって地形隆起ポイントの高さが増加された場合には、地形隆起キャラクタを高く大きく表示するとともに、地形が隆起していることを連想させるよう

な音を発生するとより効果的である。ステップ66の後、ステップ67において、ゲームオーバーか否かが判断される。例えば、所定時間を過ぎた場合にゲームオーバーにする等ゲーム内容に応じた適当な条件によってゲームオーバーの判定がされる。ステップ67において、ゲームオーバーであると判定されるとメインルーチンを終了する。ステップ67において、ゲームオーバーでないと判定されると、ステップ61に戻る。

#### 【0095】

図48は視界移動処理のフローチャートである。まず、ステップ621において、変換テーブルを参照して、スクロールカウンタX座標(SCx)およびY座標(SCy)の変更処理が行われる。ステップ621の後、ステップ622～ステップ629において、ゲーム空間の端を超えてスクロールしようとしているか否かが判断されて、ゲーム空間の端を超えてスクロールしようとしている場合には、スクロールカウンタの値(SCx、SCy)を適正な値にする処理が行われる。

#### 【0096】

ステップ622において、スクロールカウンタX座標(SCx)がスクロール限界値X座標(SCxmax)を超えているか否かが判断され、超えていないと判断された場合には、ステップ624に進む。ステップ622において、超えていると判断された場合には、ステップ623に進み、スクロールカウンタX座標(SCx)の値がスクロール限界値X座標(SCxmax)に設定された後、ステップ624に進む。

#### 【0097】

ステップ624において、スクロールカウンタX座標(SCx)が0より小さいか否かが判断され、0以上であると判断された場合は、ステップ626に進む。ステップ624において、0より小さいと判断された場合は、ステップ625に進み、スクロールカウンタX座標(SCx)の値が0に設定された後、ステップ626に進む。

#### 【0098】

ステップ626において、スクロールカウンタY座標(SCy)がスクロール

限界値Y座標（S C y m a x）を超えているか否かが判断され、超えていないとと判断された場合は、ステップ628に進む。ステップ626において、超えていると判断された場合は、ステップ627に進み、スクロールカウンタY座標（S C y）の値がスクロール限界値Y座標（S C y m a x）に設定された後、ステップ628に進む。

#### 【0099】

ステップ628において、スクロールカウンタY座標（S C y）が0より小さいか否かが判断され、0以上であると判断された場合は、視界移動処理を終了する。ステップ628において、0より小さいと判断された場合は、ステップ629に進み、スクロールカウンタY座標（S C y）の値が0に設定された後、視界移動処理を終了する。

#### 【0100】

図49は地形隆起処理のフローチャートである。まず、ステップ631において、Z軸接点スイッチの出力があったか否か（すなわち、Z軸方向の衝撃入力があったか否か）が判断され、Z軸接点スイッチの出力がなかったと判断された場合は、地形隆起処理を終了する。Z軸接点スイッチの出力があったと判断された場合は、ステップ632に進む。ステップ632において、Aボタン（操作スイッチ13b）が押されているか否かが判断され、Aボタンが押されていると判断された場合は、ステップ633に進み、LCDに表示されている領域内の地形隆起ポイントの高さ（H）をそれぞれ1増加させる処理がされる。ステップ633の後、地形隆起処理を終了する。

#### 【0101】

ステップ632において、Aボタンが押されていないことが判断されると、ステップ634に進み、Bボタン（操作スイッチ13c）が押されているか否かが判断される。Bボタンが押されていることが判断されると、ステップ635に進み、LCDに表示されている領域外の地形隆起ポイントの高さ（H）をそれぞれ1増加させる処理がされる。ステップ635の後、地形隆起処理を終了する。ステップ634において、Bボタンが押されていないことが判断されると、ステップ636において、全地形隆起ポイントの高さ（H）をそれぞれ1増加させる処

理がされた後、地形隆起処理を終了する。

#### 【0102】

##### (第3実施例)

次に、図50～図59を参照して、本発明の第3の実施例を説明する。このゲームは、ゲーム装置をフライパンや包丁であるかのように動かして仮想的な料理を楽しむゲームである。

#### 【0103】

図50～図53は、ゲーム画面の一例である。図50において、ゲーム画面には、プレイヤキャラクタ91とキッチン92とコンロ93とフライパン94と机95とまな板96が表示される。Aボタン（操作スイッチ13b）を押すと、図51および図52を参照して後述するフライパン空間処理が開始する。また、Bボタン（操作スイッチ13c）を押すと、図53を参照して後述する包丁空間処理が開始する。

#### 【0104】

図51および図52は、フライパン空間処理のゲーム画面の一例である。フライパン空間処理では、ゲーム装置をフライパンのように操作して、目玉焼きを調理するゲームをする。図51(a)において、ゲーム画面には、フライパン94と卵97が表示されている。図51(a)に示す状態において、ゲーム装置をY軸を中心にマイナス方向に傾けると、図51(b)に示すように、卵97はフライパンの左方へ移動表示される。また、図51(b)に示す状態において、ゲーム装置をX軸を中心としてプラス方向に傾けると、卵97はフライパンの下方へ移動表示される。このように処理することによって、ゲーム装置をあたかもフライパンのように操作して、卵がフライパンの傾きによって移動しているかのような感覚をプレイヤに与えることができる。

#### 【0105】

図52(a)に示す状態において、ゲーム装置のZ軸方向に衝撃入力を与えると、図52(b)に示すように、卵97はフライパン94から離れて上方にジャンプしているような表示がされ、その後、図52(c)または(d)に示すように、卵97は着地する表示がされる。このとき、図52(a)に示すように、Z

軸方向に衝撃入力をした時の卵97の位置がフライパン94の端の方である場合には、卵97はフライパン94からはみだしてジャンプし着地する（図52（c））こととなり、失敗となる。なお、図52（b）に示す状態において、ゲーム装置をスライド移動させることによって、卵97とフライパン94との相対的な位置関係を修正して、卵97をフライパン94の中に着地させることが可能である（図52（d））。このように処理することによって、ゲーム装置をあたかもフライパンのように操作して、ジャンプした卵をフライパンで受け止めるかのような感覚をプレイヤに与えることができる。

#### 【0106】

図53は包丁空間処理のゲーム画面の一例である。包丁空間処理では、ゲーム装置を包丁のように操作して、キャベツを千切りするゲームをする。図53（a）において、ゲーム画面には、包丁98とキャベツ99が表示されている。図53（a）に示す状態において、ゲーム装置をX軸のプラス方向にスライド移動させると、図53（b）に示すように、包丁98に対してキャベツ99が左に移動する表示がされる（包丁98は常にゲーム画面の中央に表示されるため、キャベツ99が相対的に左に移動する表示がされる）。このように処理することによって、ゲーム装置をあたかも包丁のように操作して、キャベツと包丁との位置関係を調節してキャベツを切断する位置を調節しているかのような感覚をプレイヤに与えることができる。

#### 【0107】

さらに、図53（b）に示す状態において、ゲーム装置を上下に運動（Z軸方向の運動入力）させると、キャベツ99が包丁98によって千切りされる表示がされる。この際、キャベツが切断される音を発生させるとより効果的である。

#### 【0108】

以下、図54を参照して、メモリに記憶されるデータを説明する。なお、プログラムROM34には、第1実施例のプログラムROM（図16）と略同様のプログラムが記憶されるが、加速度センサ出力値変換テーブル記憶領域には、フライパン用テーブルと卵ジャンプ時用テーブルと包丁用テーブルが記憶され、ゲームプログラム記憶領域には、メインプログラム、センサ出力読み取りプログラム

、フライパン空間プログラム、卵ジャンププログラム、包丁空間プログラムおよびその他のプログラムが記憶される。なお、加速度センサ出力値変換テーブルのフライパン用テーブルは、図56を参照して後述するフライパン空間プログラムで参照され、卵ジャンプ時用テーブルは、図58を参照して後述する卵ジャンププログラムで参照され、包丁用テーブルは、図57を参照して後述する包丁空間プログラムで参照される。

## 【0109】

フライパン用テーブルには、XY軸加速度センサ31の出力値（INx, INy）は卵のXおよびY座標（Ex, Ey）の変化量の計算に利用されることが定義される。これによって、ゲーム装置を傾き入力（第1の実施例における図10を参照）したときに卵の表示位置が変更され、フライパンの上を卵が滑るように表示制御される。また、座標Z軸接点スイッチ32の出力値（INz）は、卵のジャンプ判定に利用され、衝撃入力フラグ（FS）は利用されないことが定義される。

## 【0110】

卵ジャンプ時用テーブルには、XY軸加速度センサ31の出力値（INx, INy）は卵のXおよびY座標（Ex, Ey）の変化量の計算に利用されることが定義される。これによって、卵がジャンプしている間にゲーム装置をスライド入力（第1の実施例における図9を参照）したときに卵の表示位置が変更され、フライパンと卵の相対位置が変更されるように表示制御される。なお、卵ジャンプ時用テーブルにおいて補正比率はマイナス値が定義される。なぜなら、本実施例では、フライパンはゲーム画面に固定的に表示され、卵がフライパンに対して相対的に移動するように表示されるため、ゲーム装置のスライド移動の方向とは逆の方向に卵を移動表示する必要があるためである。また、Z軸接点スイッチ32の出力値（INz）および衝撃入力フラグ（FS）は利用されない。

## 【0111】

包丁用テーブルには、XY軸加速度センサ31の出力値（INx, INy）はキャベツのXおよびY座標（CAx, CAy）の変化量の計算に利用されることが定義されている。これによって、ゲーム装置をスライド入力したときにキャベ

ツの表示位置が変更され、キャベツと包丁の相対位置が変更されるように表示制御される。なお、包丁用テーブルは卵ジャンプ時用テーブルと同様に補正比率はマイナス値が定義される。なぜなら、本実施例では、包丁はゲーム画面に固定的に表示され、キャベツが包丁に対して相対的に移動するように表示されるため、ゲーム装置のスライド移動の方向とは逆の方向にキャベツを移動表示する必要があるためである。また、Z軸接点スイッチ32の出力値(INz)は、キャベツの切断処理の判定に利用され、衝撃入力フラグ(FS)は利用されないことが定義される。

#### 【0112】

図54はワークRAM26のメモリマップである。ワークRAM26には、CPU21がゲームプログラムを実行する際の一時的なデータが記憶される。具体的には、加速度センサ出力値記憶領域263a、衝撃入力フラグ記憶領域263b、卵データ記憶領域263cおよびキャベツデータ記憶領域263dが含まれる。

#### 【0113】

加速度センサ出力値記憶領域263aおよび衝撃入力フラグ記憶領域263bに記憶されるデータは第1の実施例と同様であるので説明を省略する。

卵データ記憶領域263cには、卵のX座標(Ex)、Y座標(Ey)、高さ(Eh)および焼け具合(Ef)のデータが記憶される。キャベツデータ記憶領域263dには、キャベツのX座標(CAx)、Y座標(CAy)および切断割合(CAc)のデータが記憶される。

表示用RAMのメモリマップは第1の実施例における図18と同様であるので説明を省略する。

#### 【0114】

以下、図55～図59を参照して、ゲームプログラムの処理の流れを説明する。

図55はメインルーチンのフローチャートである。ゲーム装置本体10にカトリッジ30を差して、ゲーム装置本体10の電源をONにすると、図55に示すメインルーチンが開始される。第3の実施例においても、第1の実施例と同じ

ように、OGポジション設定処理やニュートラルポジションの設定処理をおこなっても良いが、説明を簡単にするため省略することにする。

#### 【0115】

まずステップ71において、第1の実施例における図31と同様のセンサ出力読取処理が行われて、XY軸加速度センサ31およびZ軸接点スイッチ32の出力値がセンサインターフェース33を介して読取られる（OGポジションデータおよびニュートラルポジションデータによる補正は省略）。ステップ71の後、ステップ72において、Aボタン（操作スイッチ13b）が押されているか否かが判断される。ステップ72において、Aボタンが押されていることが判断されると、ステップ73に進み、図57を参照して後述する包丁空間処理が行われた後、ステップ76に進む。

#### 【0116】

ステップ72において、Aボタンが押されていないことが判断されると、ステップ74に進み、Bボタン（操作スイッチ13c）が押されているか否かが判断される。ステップ74において、Bボタンが押されていないことが判断されると、ステップ76に進む。ステップ74において、Bボタンが押されていることが判断されると、ステップ75に進み、図56を参照して後述するフライパン空間処理が行われた後、ステップ76に進む。

#### 【0117】

ステップ76において、ゲームオーバーか否かが判断される。具体的には、例えば、所定時間を過ぎた場合にゲームオーバーにする等ゲーム内容に応じた適当な条件によってゲームオーバーの判定がされる。ステップ76において、ゲームオーバーでないと判断されると、ステップ71に戻る。ステップ76において、ゲームオーバーであると判断されると、メインルーチンを終了する。

#### 【0118】

図56はフライパン空間処理のフローチャートである。まず、ステップ771において、フライパン用テーブルを参照して、卵X座標（Ex）および卵Y座標（Ey）の変更処理が行われる。ステップ771の後、ステップ772において、図58を参照して後述する卵ジャンプ処理が行われる。ステップ772の後、

ステップ773において、卵焼け具合（E f）を1増加させる処理がおこなわれる。ステップ773の後、ステップ774において、卵焼け具合（E f）が100以上になったか否かが判断される。卵焼け具合（E f）が100より小さいことが判断されると、フライパン空間処理を終了する。卵焼け具合（E f）が100以上になったことが判断されると、ステップ775に進み、卵成功処理が行なわれる。卵成功処理では、例えば、卵の調理が完成した画面が表示され、得点が加算される処理が行われる。ステップ775の後、フライパン空間処理を終了する。

#### 【0119】

図57は包丁空間処理のフローチャートである。まず、ステップ741において、包丁用テーブルを参照して、キャベツX座標（CAx）およびキャベツY座標（CAy）の変更処理が行われる。ステップ741の後、ステップ742において、図59を参照して後述するキャベツ切断処理が行われる。ステップ742の後、ステップ743において、キャベツ切断割合（CAC）が100以上になったか否かが判断される。キャベツ切断割合（CAC）が100より小さいことが判断されると、包丁空間処理を終了する。キャベツ切断割合（CAC）が100以上になったことが判断されると、ステップ744に進み、キャベツ成功処理が行なわれる。キャベツ成功処理では、例えば、キャベツの切断が完成した画面が表示され、得点が加算される処理が行われる。ステップ744の後、包丁空間処理を終了する。

#### 【0120】

図58は卵ジャンプ処理のフローチャートである。まず、ステップ772aにおいて、Z軸接点スイッチの出力があったか否か（すなわち、Z軸方向の衝撃入力があったか否か）が判断される。ステップ772aにおいて、Z軸接点スイッチの出力がなかったことが判断されると、卵ジャンプ処理を終了する。ステップ772aにおいて、Z軸接点スイッチの出力があったことが判断されると、ステップ772bにおいて、卵がジャンプしている表示が行われる。ステップ772bの後、ステップ772cにおいて、卵の高さ（Eh）がCH（所定値）に設定される。ステップ772cの後、ステップ772dにおいて、第1の実施例にお

ける図31と同様のセンサ出力読取処理が行われて、XY軸加速度センサ31およびZ軸接点スイッチ32の出力値がセンサインターフェース33を介して読み取られる（0Gポジションデータおよびニュートラルポジションデータによる補正は省略）。ステップ772dの後、ステップ772eにおいて、卵ジャンプ時用テーブルを参照して、卵X座標（Ex）および卵Y座標（Ey）の変更処理が行われる。ステップ772eの後、ステップ772fにおいて、卵の高さ（Eh）を1減少させる処理が行われる。ステップ772fの後、ステップ772gにおいて、卵のX座標（Ex）、Y座標（Ey）、高さ（Eh）に基づいて表示処理が行われる。ステップ772gの後、ステップ772hにおいて、卵が着地したか否か、すなわち卵の高さ（Eh）が0になったか否かが判断される。ステップ772hにおいて、卵が着地していないことが判断されると、ステップ772dに戻る。ステップ772hにおいて、卵が着地したことが判断されると、ステップ772iにおいて、卵の着地位置がフライパン内か否かが判断され、フライパン内であると判断された場合にはステップ772jにおいて、ジャンプ成功処理が行われた後、卵ジャンプ処理を終了する。ジャンプ成功処理では、例えば「成功」と表示しつつ成功の音楽を発生し、得点を加算する処理が行われる。ステップ772iにおいて、卵の着地位置がフライパンの外であると判断された場合には、ステップ772kにおいて、ジャンプ失敗処理が行われた後、卵ジャンプ処理を終了する。ジャンプ失敗処理では、例えば「失敗」と表示しつつ失敗の音楽を発生し、卵焼け具合（Ef）を0にする処理が行われる（卵の調理をやり直させるようにする）。

## 【0121】

図59はキャベツ切断処理のフローチャートである。まず、ステップ742aにおいて、Z軸接点スイッチの出力があったか否か（すなわち、Z軸方向の運動入力があったか否か）が判断される。ステップ742aにおいて、Z軸接点スイッチの出力がなかったことが判断されると、キャベツ切断処理を終了する。ステップ742aにおいて、Z軸接点スイッチの出力があったことが判断されると、ステップ742bにおいて、包丁の下にキャベツがあるか否かが判断される。ステップ742bにおいて、包丁の下にキャベツがないことが判断されると、キャ

ベツ切断処理を終了する。ステップ742bにおいて、包丁の下にキャベツがあることが判断されると、ステップ742cにおいて、表示処理（キャベツが一定量切断される表示）がされる。ステップ742cの後、ステップ742dにおいて、キャベツの切断割合（C A c）を1増加する処理がされた後、キャベツ切断処理を終了する。

## 【0122】

## (第4実施例)

次に、図60～図66を参照して、本発明の第4の実施例を説明する。図60はゲーム空間の概念図および複数のゲーム装置のゲーム画面の一例を示した図である。このゲームは、複数のゲーム装置で通信することによってゲーム空間を共有して、第1の実施例と同様のゲームを複数のプレイヤでおこなって対戦（または協力）を楽しむゲームである。ゲーム空間である迷路板は複数のゲーム装置10および40に共通であり、ゲーム装置10のゲーム画像とゲーム装置40のゲーム画像は同じゲーム空間データに基づいている（ただし、それぞれのゲーム装置で視界が異なる）。第1のゲーム装置10のLCDには1点鎖線で示される範囲12が表示され、第2のゲーム装置40のLCDには点線で示される範囲42が表示される。第1の実施例と同様に、ゲーム装置の傾きに応じて、ゲーム空間である迷路板が傾いたことがシミュレートされるが、本実施例の場合は、ゲーム装置10の傾きとゲーム装置40の傾きを合成した値によって迷路板の傾きがシミュレートされる（一方のゲーム装置の傾きのみによって迷路版の傾きをシミュレートしてもよい）。ゲーム装置10のプレイヤは自分のボール61aを操作するためにゲーム装置10を傾けて迷路板の傾きを操作しようとするが、一方でゲーム装置40のプレイヤは自分のボール61bを操作するためにゲーム装置40を傾けて迷路板の傾きを操作しようとするので、お互いに自分の思う通りには迷路板の傾きを操作することができず、より複雑なゲームを楽しむことができる。なお、本実施例においては、通信ケーブル50を用いて2つのゲーム装置間の通信をおこなうこととするが、無線や携帯電話等の通信手段を利用しても良い。

## 【0123】

第4の実施例のプログラムROMには、第1の実施例のプログラムROM（図

16)と略同様のデータが記憶されるが、ゲームプログラム記憶領域には、第1の実施例の場合に追加して、図63および図64を参照して後述するマップ確認プログラムおよび図65および図66を参照して後述する通信割込プログラムがさらに記憶される。

#### 【0124】

ゲームプログラム記憶領域に記憶されるプログラムのうち、メインプログラムとマップ確認プログラムと通信割込プログラムは、ゲーム装置10とゲーム装置40とで異なったプログラムとなっている。これは、ゲーム装置10を親機とし、ゲーム装置40を子機として通信処理を行うためであるが、詳細については、図61～図66を参照して後述する。

#### 【0125】

第4の実施例のワークRAMには、第1の実施例のワークRAM17と略同様のデータが記憶されるが、第1の実施例の場合に追加して、合成データ記憶領域をさらに含む。合成データ記憶領域には、ゲーム装置10のXY軸加速度センサ31およびZ軸接点スイッチ32の出力値と、ゲーム装置40のXY軸加速度センサ31およびZ軸接点スイッチ32の出力値を合成した値がそれぞれ記憶される。

表示用RAMおよびバックアップRAMのメモリマップは第1の実施例における図18および図19と同様であるので説明を省略する。

#### 【0126】

以下、図61～図66を参照して、ゲームプログラムの処理の流れを説明する。

図61はゲーム装置10で実行されるメインルーチンのフローチャートである。本実施例においては、説明を簡単にするために、0G設定処理、ニュートラルポジション設定処理および衝撃入力による波発生処理を省略しているが、第1の実施例と同様に、これらの処理を追加しても良い。

#### 【0127】

まず、ステップ81pにおいて、第1の実施例における図30と同様のゲームマップ選択処理が行われる。ステップ81pの後、ステップ82pにおいて、図

63を参照して後述する親機マップ確認処理が行われる。ステップ82pの後、ステップ83pに進む。

#### 【0128】

ステップ83pからステップ85pまでがメインループであり、ゲームオーバーになるかまたはゲームクリアになるまで繰り返し処理される。ステップ83pにおいて、ワークRAM26のデータに基づいて表示用RAM25に必要なデータが書き込まれ、表示用RAM25に記憶されたデータに基づいてLCD12にゲーム画面が表示される。ステップ84pにおいて、第1の実施例における図32から図36と同様の各オブジェクト移動処理が行われ（波移動処理は省略）、プレイヤキャラクタおよびNPCの移動処理が行われる。ステップ84pの後、ステップ85pにおいて、第1の実施例における図37と同様の衝突処理が行われ、プレイヤキャラクタとNPC等との衝突処理が行われる。ステップ85pの後、ステップ86pにおいて第1の実施例における図40と同様の画面スクロール処理が行われる。

#### 【0129】

図62はゲーム装置40で実行されるメインルーチンのフローチャートである。本実施例においては、説明を簡単にするために、OG設定処理、ニュートラルポジション設定処理および衝撃入力による波発生処理を省略しているが、第1の実施例と同様にこれらの処理を追加しても良い。

#### 【0130】

まず、ステップ81cにおいて、第1の実施例における図30と同様のゲームマップ選択処理が行われる。ステップ81cの後、ステップ82cにおいて、図64を参照して後述する子機マップ確認処理が行われる。ステップ82cの後、ステップ83cに進む。

#### 【0131】

ステップ83cからステップ88cまでがemainループであり、ゲームオーバーになるかまたはゲームクリアになるまで繰り返し処理される。まず、ステップ83cにおいて、ワークRAM26のデータに基づいて表示用RAM25に必要なデータが書き込まれ、表示用RAM25に記憶されたデータに基づいてLCD1

2にゲーム画面が表示される。ステップ83cの後、ステップ84cにおいて、第1の実施例における図31と同様のセンサ出力読取処理が行われて、XY軸加速度センサ31およびZ軸接点スイッチ32の出力値がセンサインターフェース33を介して読取られる（0Gポジションデータおよびニュートラルポジションデータによる補正是省略）。ステップ84cの後、ステップ85cにおいて、割込信号および先のステップ84cにおいて読込まれてワークRAM26に記憶された加速度センサ出力値データ（INx, INy, INz）がゲーム装置10に送信される。ゲーム装置10側では、この割込信号を受けて、図65を参照して後述する親機通信割込処理が開始される。ステップ85cの後、ステップ86cにおいて、第1の実施例における図32から図36と同様の各オブジェクト移動処理が行われ（波移動処理は省略）、プレイヤキャラクタおよびNPCの移動処理が行われる。ステップ86cの後、ステップ87cにおいて、第1の実施例における図37と同様の衝突処理が行われ、プレイヤキャラクタとNPC等との衝突処理が行われる。ステップ87cの後、ステップ88cにおいて第1の実施例における図40と同様の画面スクロール処理が行われる。

#### 【0132】

図63は、ゲーム装置10で実行される親機マップ確認処理のフローチャートである。まず、ステップ87p1において、自身のワークRAM26に記憶されたマップナンバーデータがゲーム装置40に送信される。ステップ87p1の後、ステップ87p2において、データの受信がおこなわれる。具体的には、図64を参照して後述する子機マップ確認処理のステップ87c3においてゲーム装置40から送信されるマップナンバーデータを受信する。ステップ87p3において、データが受信されたことが判断されると、ステップ87p4において、自身のマップナンバーデータと先のステップ87p2において受信したゲーム装置40のマップナンバーデータが一致するか否かが判断される。ステップ87p4において、マップナンバーデータが一致することが判断されると、親機マップ確認処理を終了する。ステップ87p4において、マップナンバーデータが一致しないことが判断されると、図61のメインルーチンのステップ81pのゲームマップ選択処理に戻る。

## 【0133】

図64は、ゲーム装置40で実行される子機マップ確認処理のフローチャートである。まず、ステップ87c1において、データの受信が行われる。具体的には、前述の図63の親機マップ確認処理のステップ87p1においてゲーム装置10から送信されるマップナンバーデータを受信する。ステップ87c2においてデータが受信したことが判断されると、ステップ87c3において、自身のワークRAM26に記憶されたマップナンバーデータがゲーム装置10に送信される。ステップ87c3の後、ステップ87c4において、自身のマップナンバーデータと先のステップ87c1において受信したゲーム装置10のマップナンバーデータが一致するか否かが判断される。ステップ87c4において、マップナンバーデータが一致することが判断されると、子機マップ確認処理を終了する。ステップ87c4において、マップナンバーデータが一致しないことが判断されると、図62のメインルーチンのステップ81cのゲームマップ選択処理に戻る。

## 【0134】

図65は、ゲーム装置10で実行される親機通信割込み処理のフローチャートである。この処理は、前述の図62に示すゲーム装置40のメインルーチンのステップ85cにおいて送信される割込信号によって処理が開始される。まず、ステップ91pにおいて、データの受信を行う。具体的には、前述の図62に示すゲーム装置40のメインルーチンのステップ85cにおいて送信されるゲーム装置40の加速度センサ出力値を受信する。ステップ91pの後、ステップ92pにおいて、第1の実施例における図31と同様のセンサ出力読取処理が行われて、XY軸加速度センサ31およびZ軸接点スイッチ32の出力値がセンサインターフェース33を介して読取られる（OGポジションデータおよびニュートラルポジションデータによる補正は省略）。ステップ92pの後、ステップ93pにおいて、先のステップ91pで受信したゲーム装置40の加速度センサ出力値と先のステップ92pで読取ったゲーム装置10の加速度センサ出力値を合成する。ここで、合成とは、単に加算する計算処理をおこなっても良いし、例えば2つの値に重みをつけて加算する等、より複雑な計算式により2つの値から合成値を

算出しても良い。ステップ93pの後、ステップ94pにおいて、割込信号および先のステップ93pで計算された合成データがゲーム装置40に送信される。

#### 【0135】

図66は、ゲーム装置40で実行される子機通信割込み処理のフローチャートである。この処理は、図65の親機通信割込み処理のステップ94pにおいて送信される割込信号に応じて処理が開始される。ステップ91cにおいてゲーム装置10から合成データを受信して終了する。

#### 【0136】

上述の実施例は、携帯型ゲーム装置が検出手段を備えるものであったが、図67に示すように家庭用ゲーム機、パソコンまたは業務用ゲーム機等のコントローラが検出手段を備えるようにしてもよい。この場合には、プレイヤはコントローラを傾けたり、運動や衝撃を与えることによってテレビジョン受信機等の表示装置に表示されるゲーム空間の制御をおこなう。例えば、図68に示すように、コントローラを傾けることによって表示装置に表示されたゲーム空間である板が傾く表示がされ、かつ、板の上のボールが転がることがシミュレートされる。コントローラを右に傾けると板が右に傾いてボールが右に転がり、コントローラを左に傾けると板が左に傾いてボールが左に転がることがシミュレートされる。

#### 【0137】

上述の実施例では、カートリッジに加速度センサを設けたが、携帯型ゲーム装置本体側に加速度センサを設けても良い。携帯型ゲーム装置本体側に加速度センサを設けた場合、カートリッジ毎に加速度センサを備える必要がなくコストを削減できる。また、ゲーム装置に用いる情報記憶媒体は、カートリッジに限るものではなく、PCカードのようなICカードであっても良い。

#### 【0138】

上述の第1の実施例では、ニュートラルポジションデータはワークRAM26に記憶してゲームプレイ毎に設定するようにしたが、バックアップRAM35に記憶して次回のゲームプレイにおいても同じデータが利用可能にしても良い。

#### 【0139】

上述の第1の実施例では、ニュートラルポジションはプレイヤが決定するよう

にしたが、予めゲームプログラムにニュートラルポジションデータを記憶していてこれを利用するようにしても良い。また、複数のニュートラルポジションデータを記憶しておいて、プレイヤがいずれか一つを選択するようにしても良い。

## 【0140】

上述の第1の実施例では、ゲームキャラクタは、プレイヤキャラクタ（ボール）と敵キャラクタ（亀）のみであったが、これらに加えて、プレイヤキャラクタを助ける味方キャラクタや中立的なキャラクタ等のN P C（ノンプレイヤーキャラクタ）を登場させても良い。これらのN P Cは、ゲームプログラムに基づいて自律移動されるが（自律移動しないN P Cがあっても良い）、プレイヤによる操作（傾き、運動または衝撃入力）に応じて移動や変形等させても良い。

## 【0141】

上述の第1の実施例では、ゲーム空間の制御は加速度センサの出力のみに基づいているが、操作スイッチに基づいてゲーム空間を制御する部分があってもよい。例えば、ピンボールゲームにおいて、ゲーム装置を傾けたり揺らしたりすることによってゲーム空間であるピンボール台を制御しつつ、操作スイッチを押した場合にフリッパーが動作するようなゲームが考えられる。また、いわゆる「落ちゲー」といわれる、落下する物体を積み上げて、積み上げ状態に応じて得点を計算するようなゲームにおいて、ゲーム装置を傾けたり揺らしたりすることによってゲーム空間を制御しつつ、操作スイッチで物体の向きを変えたり、衝撃入力で物体を高速移動させたり、Z軸方向の運動入力によって物体を変形させたりするようなゲームが考えられる。

## 【0142】

上述の第1の実施例では、ゲームキャラクタはゲーム装置の傾き（すなわち、ゲーム空間である迷路板の傾き）に応じて移動することとしたが、ゲーム装置の運動または衝撃に応じて移動するようにしてもよい。例えば、ゲーム装置をスライド移動させたときに、迷路板の壁が同じように移動したことがシミュレートされて、壁に接しているゲームキャラクタが壁に押されたかのように移動するような表示制御をおこなうことが考えられる。

## 【0143】

上述の第1の実施例では、プレイヤキャラクタ（ボール）自体を移動表示したが、プレイヤキャラクタは固定的に表示し、ゲーム空間をスクロール表示してプレイヤキャラクタがゲーム空間を相対的に移動している表示処理としてもよい。

#### 【0144】

上述の第4の実施例は、2人のプレイヤが迷路板を傾けるという同一の制御を行うものであったが、2人のプレイヤが別々の制御を担当するようにしても良い。例えば、一人のプレイヤはゲーム装置を傾けることによって迷路板を傾ける制御をし、他方のプレイヤはゲーム装置にZ軸方向の運動入力をしてゲームキャラクタをジャンプさせたり、XY軸方向の衝撃入力をして波を発生する制御をする等のゲームが考えられる。

#### 【0145】

上述の第4の実施例は、メインプログラム、マップ確認プログラムおよび通信割り込みプログラムについて、ゲーム装置10には親機用のプログラムが記憶され、ゲーム装置40には子機用のプログラムが記憶されていたが、ゲーム装置10およびゲーム装置40のそれぞれに親機用のプログラムと子機用のプログラムの両方を記憶しておいて、ゲーム開始に先立って、どちらを親機とし子機とするかを設定し、設定に従ってプログラムを選択するようにしてもよい。

#### 【図面の簡単な説明】

##### 【図1】

本発明の一実施例の携帯型ゲーム装置の外観図である。

##### 【図2】

XYZ軸の定義を示した図である。

##### 【図3】

携帯型ゲーム装置のブロック図である。

##### 【図4】

センサインターフェースのブロック図である。

##### 【図5】

加速度センサの出力を計測する原理を示した図である。

##### 【図6】

Z軸接点スイッチの構造を示した図である。

【図7】

Z軸接点スイッチがZ軸方向の運動入力（または衝撃入力）を検出する場合の図である。

【図8】

第1の実施例のゲーム画面の一例である。

【図9】

スライド入力を示した図である。

【図10】

傾き入力を示した図である。

【図11】

X軸方向またはY軸方向の衝撃入力を示した図である。

【図12】

Z軸方向の運動入力（衝撃入力）を示した図である。

【図13】

スライド入力の利用方法を示した図である。

【図14】

傾き入力の利用方法を示した図である。

【図15】

衝撃入力の利用方法を示した図である。

【図16】

第1の実施例のプログラムROMのメモリマップである。

【図17】

第1の実施例のワークRAMのメモリマップである。

【図18】

第1の実施例の表示用RAMのメモリマップである。

【図19】

第1の実施例のバックアップRAMのメモリマップである。

【図20】

第1の実施例の加速度センサ出力変換テーブルである。

【図21】

第1の実施例の加速度センサ出力変換テーブルである。

【図22】

第1の実施例の加速度センサ出力変換テーブルである。

【図23】

第1の実施例の加速度センサ出力変換テーブルである。

【図24】

第1の実施例の加速度センサ出力変換テーブルである。

【図25】

第1の実施例の加速度センサ出力変換テーブルである。

【図26】

第1の実施例の加速度センサ出力変換テーブルである。

【図27】

第1の実施例のメインルーチンのフローチャートである。

【図28】

第1の実施例の0G設定処理のフローチャートである。

【図29】

第1の実施例のニュートラルポジション設定処理のフローチャートである。

【図30】

第1の実施例のゲームマップ選択処理のフローチャートである。

【図31】

第1の実施例のセンサ出力読取処理のフローチャートである。

【図32】

第1の実施例の各オブジェクト移動処理のフローチャートである。

【図33】

第1の実施例のプレイヤキャラクタ移動処理のフローチャートである。

【図34】

第1の実施例のNPC移動処理のフローチャートである。

【図35】

第1の実施例のジャンプ移動処理のフローチャートである。

【図36】

第1の実施例の波移動処理のフローチャートである。

【図37】

第1の実施例の衝突処理のフローチャートである。

【図38】

第1の実施例の画面スクロールの説明図（スクロール前）である。

【図39】

第1の実施例の画面スクロールの説明図（スクロール後）である。

【図40】

第1の実施例の画面スクロール処理のフローチャートである。

【図41】

第2の実施例のゲーム画面の一例である。

【図42】

第2の実施例のゲーム画面（地形隆起処理）の一例である。

【図43】

第2の実施例のゲーム画面（視界移動処理）の一例である。

【図44】

第2の実施例のゲーム画面（温度上昇処理）の一例である。

【図45】

第2の実施例のプログラムROMのメモリマップである。

【図46】

第2の実施例のワークRAMのメモリマップである。

【図47】

第2の実施例のメインルーチンのフローチャートである。

【図48】

第2の実施例の視界移動処理のフローチャートである。

【図49】

第2の実施例の地形隆起処理のフローチャートである。

【図50】

第3の実施例のゲーム画面の一例である。

【図51】

第3の実施例のゲーム画面（フライパン空間処理）の一例である。

【図52】

第3の実施例のゲーム画面（フライパン空間処理）の一例である。

【図53】

第3の実施例のゲーム画面（包丁空間処理）の一例である。

【図54】

第3の実施例のワークRAMのメモリマップである。

【図55】

第3の実施例のメインルーチンのフローチャートである。

【図56】

第3の実施例のフライパン空間処理のフローチャートである。

【図57】

第3の実施例の包丁空間処理のフローチャートである。

【図58】

第3の実施例の卵ジャンプ処理のフローチャートである。

【図59】

第3の実施例のキャベツ切断処理のフローチャートである。

【図60】

第4の実施例のゲーム画面の一例である。

【図61】

第4の実施例のゲーム装置10のメインルーチンのフローチャートである。

【図62】

第4の実施例のゲーム装置40のメインルーチンのフローチャートである。

【図63】

第4の実施例の親機マップ確認処理のフローチャートである。

【図64】

第4の実施例の子機マップ確認処理のフローチャートである。

【図65】

第4の実施例の親機通信割込み処理のフローチャートである。

【図66】

第4の実施例の子機通信割込み処理のフローチャートである。

【図67】

本発明を家庭用ゲーム装置のコントローラに適用した場合の例である。

【図68】

本発明を家庭用ゲーム装置のコントローラに適用した場合の画面例である。

【符号の説明】

10：ゲーム装置本体

12：LCD

13：操作スイッチ

21：CPU

25：表示用RAM

26：ワークRAM

30：ゲームカートリッジ

31：XY軸加速度センサ

32：Z軸接点スイッチ

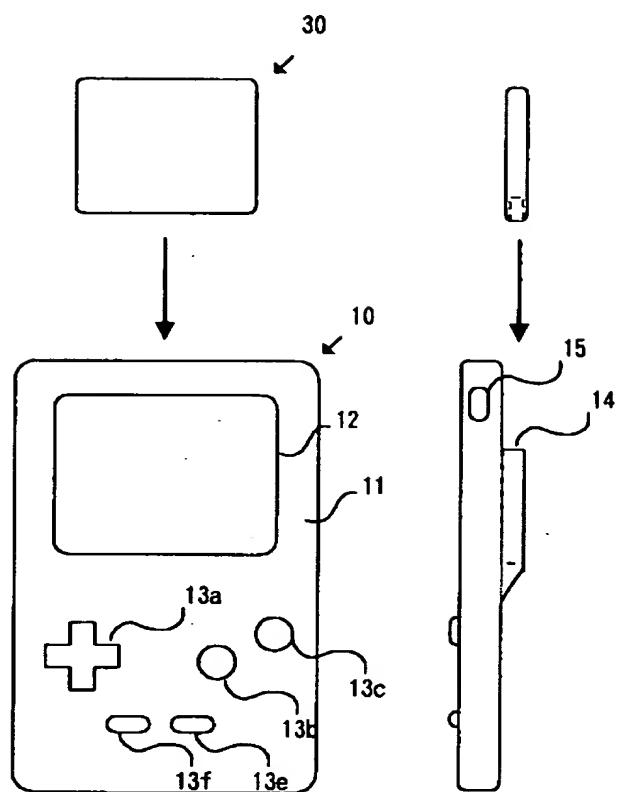
33：センサインターフェース

34：プログラムROM

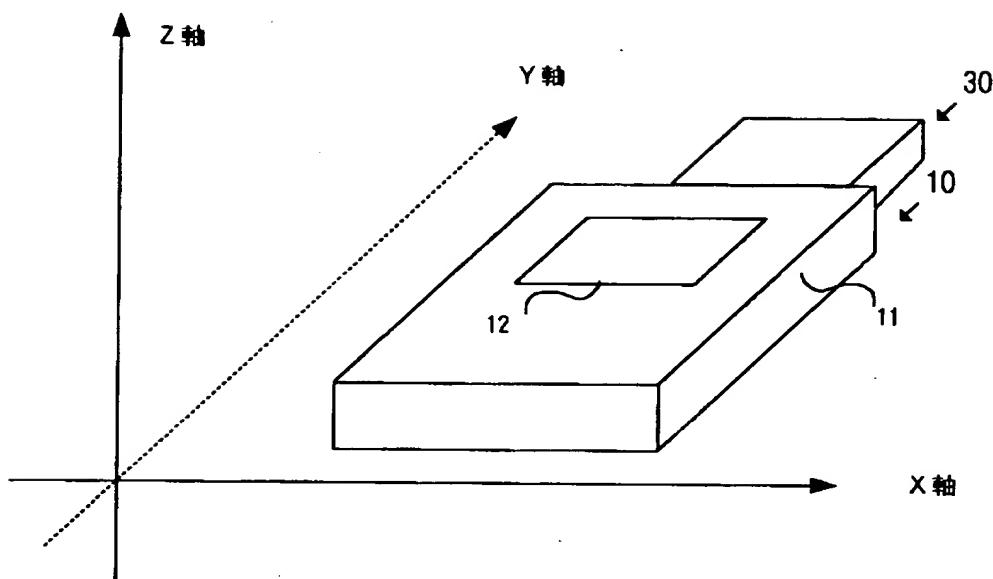
35：バックアップRAM

【書類名】 図面

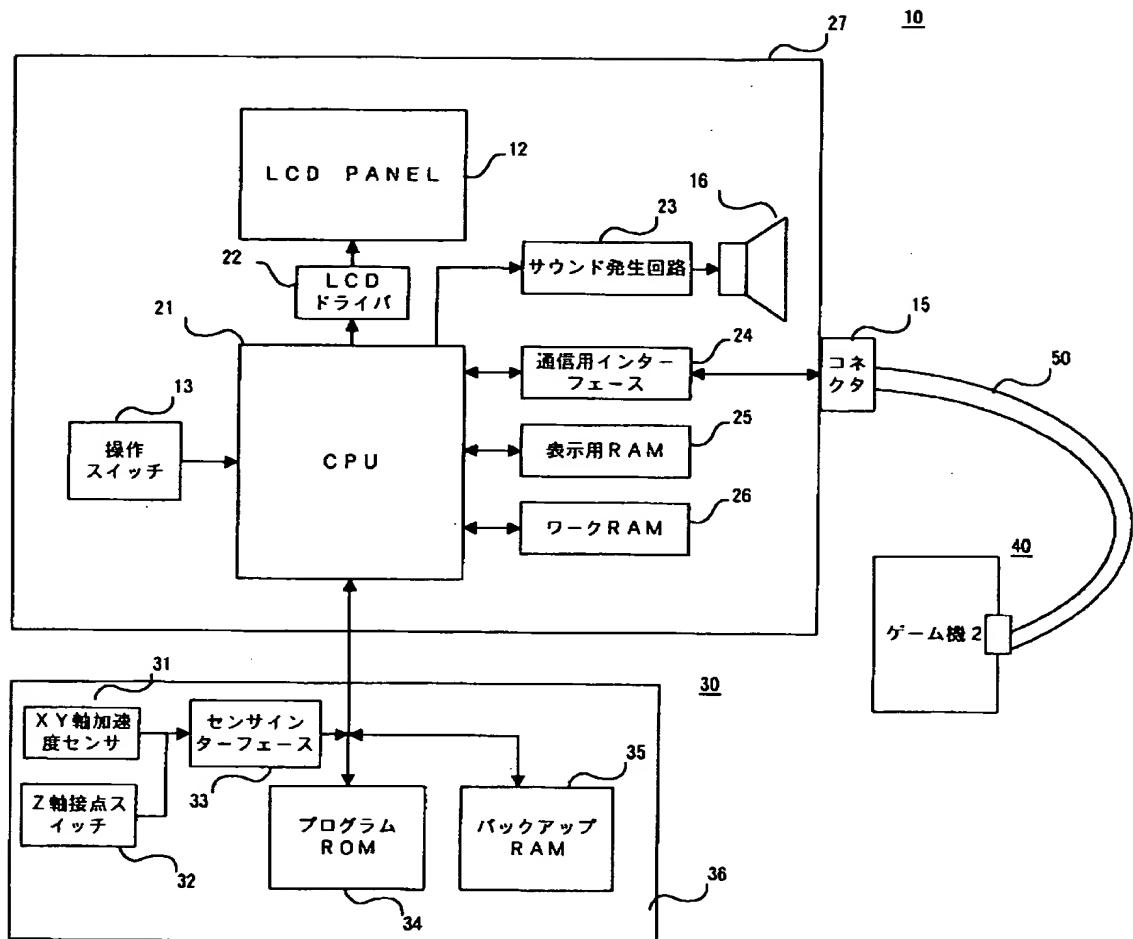
【図1】



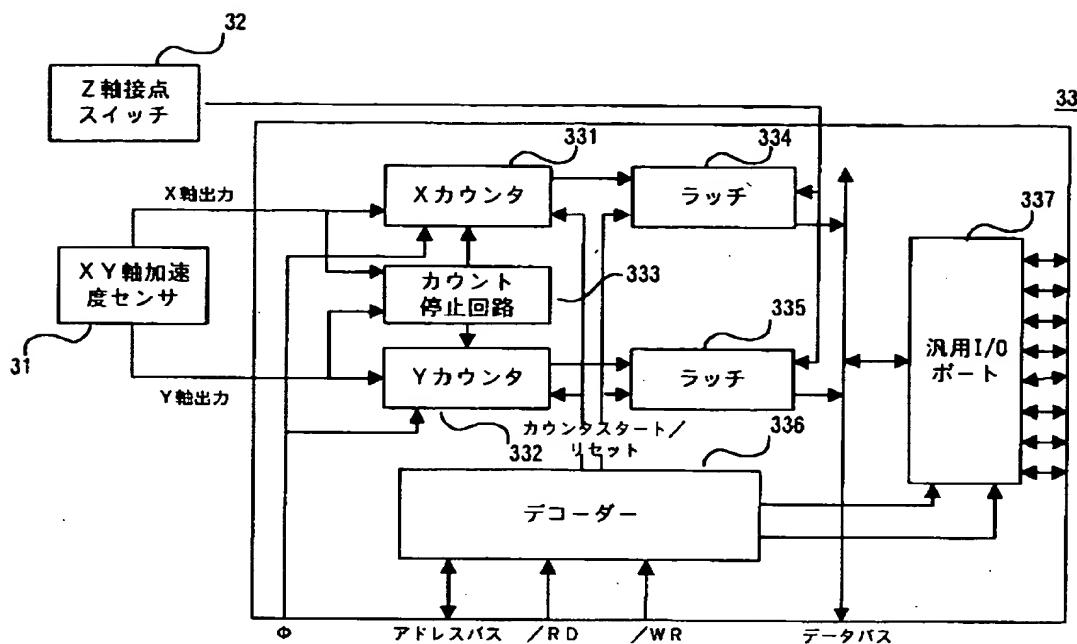
【図2】



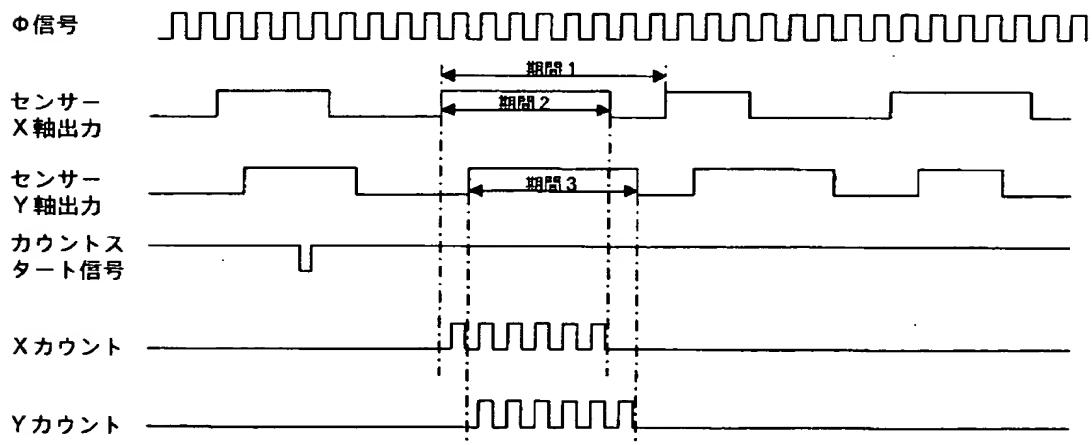
【図3】



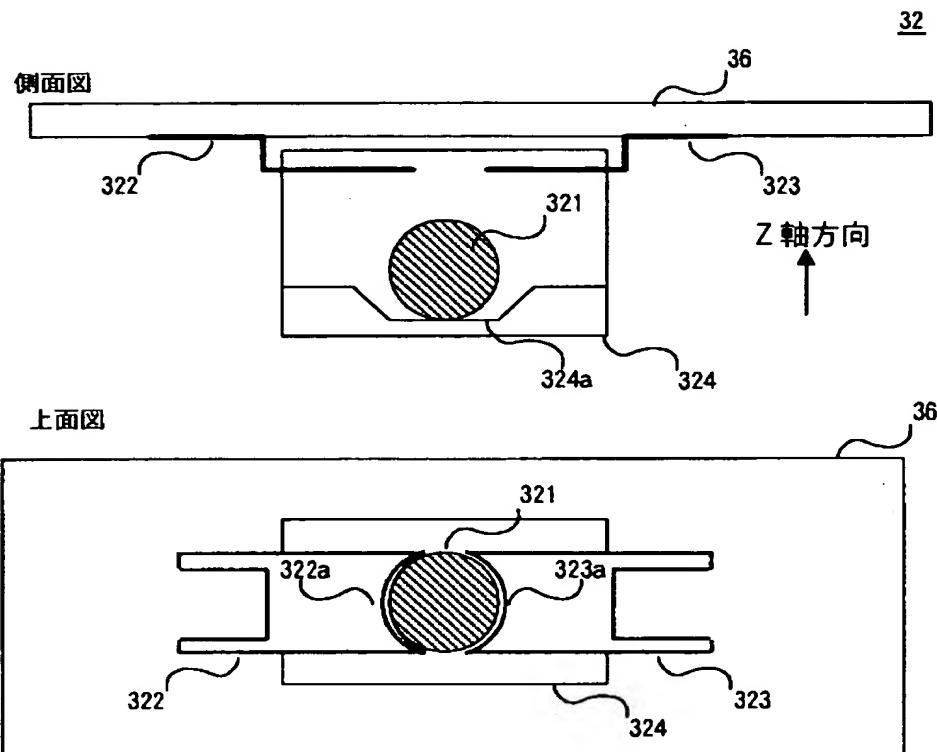
【図4】



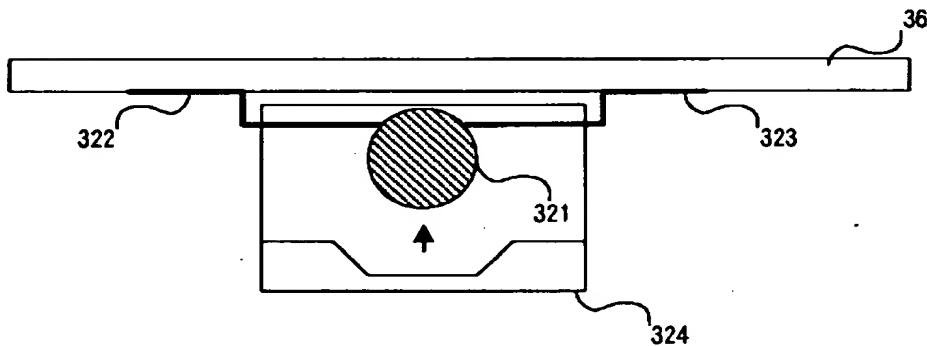
【図5】



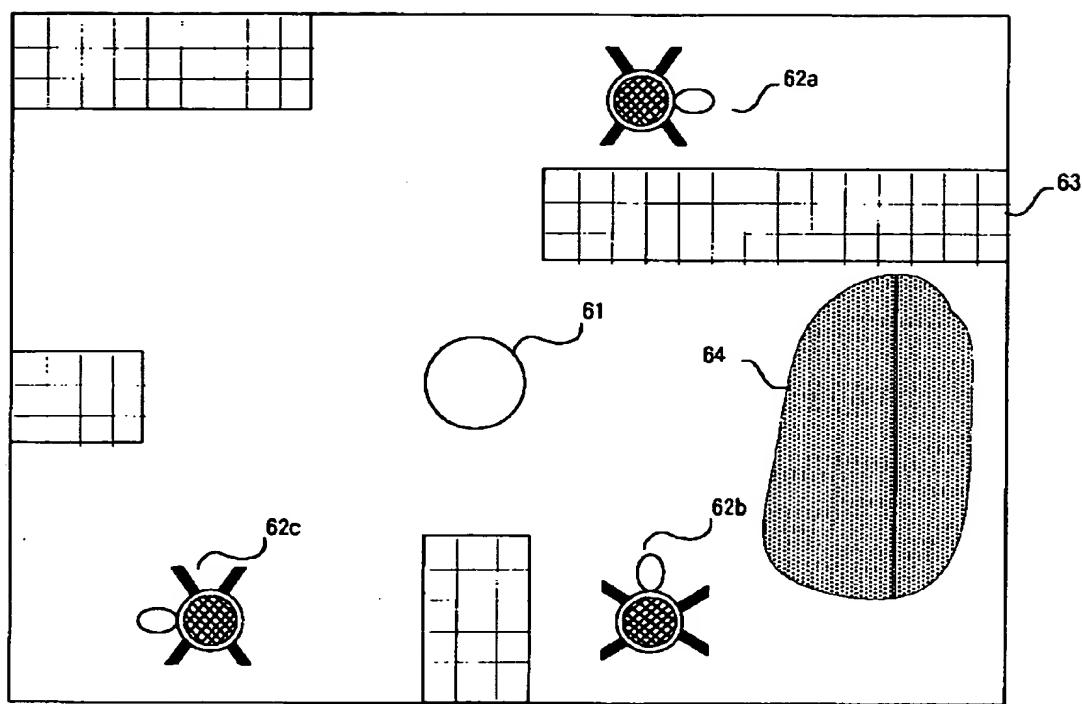
【図6】



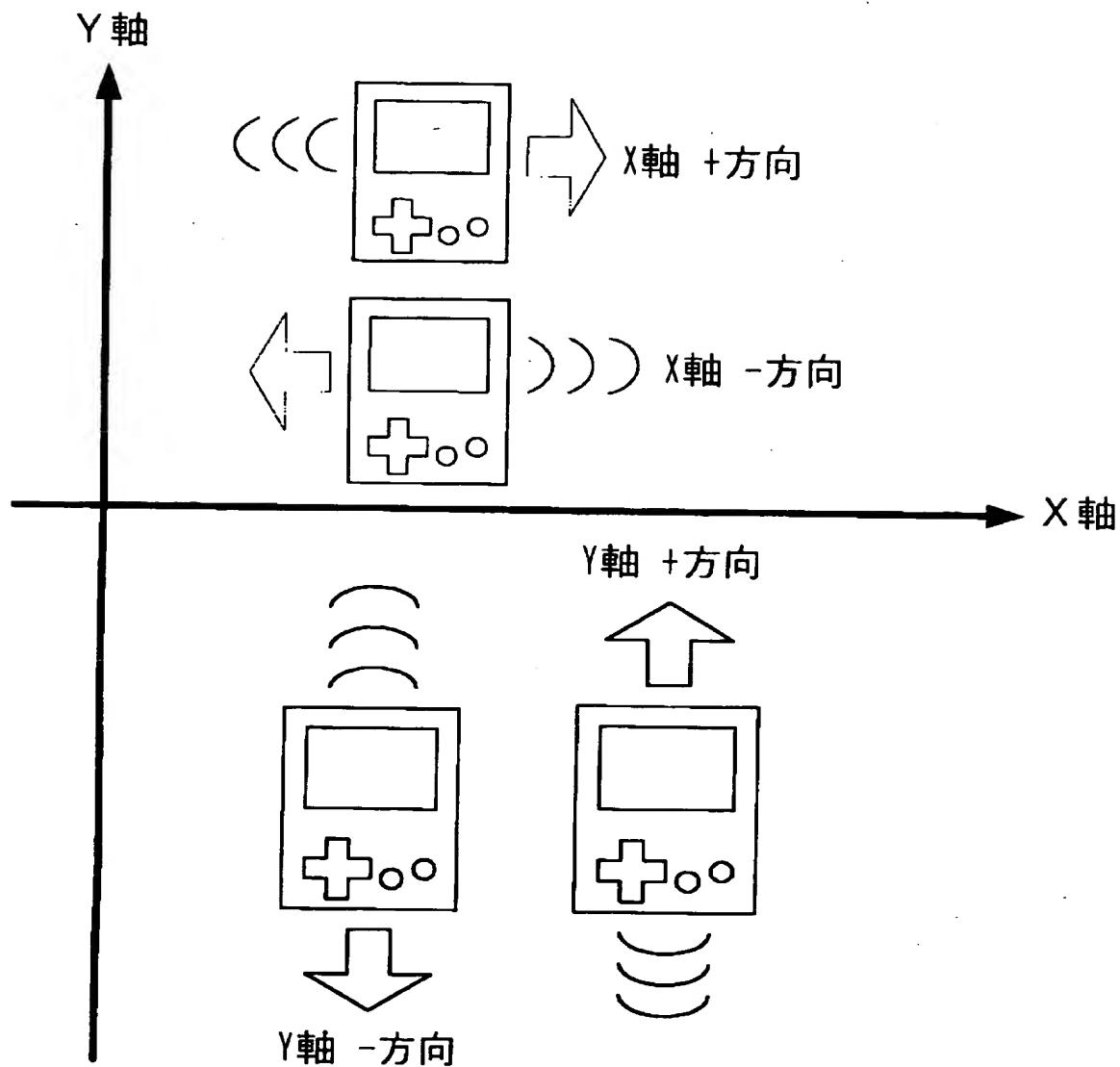
【図7】



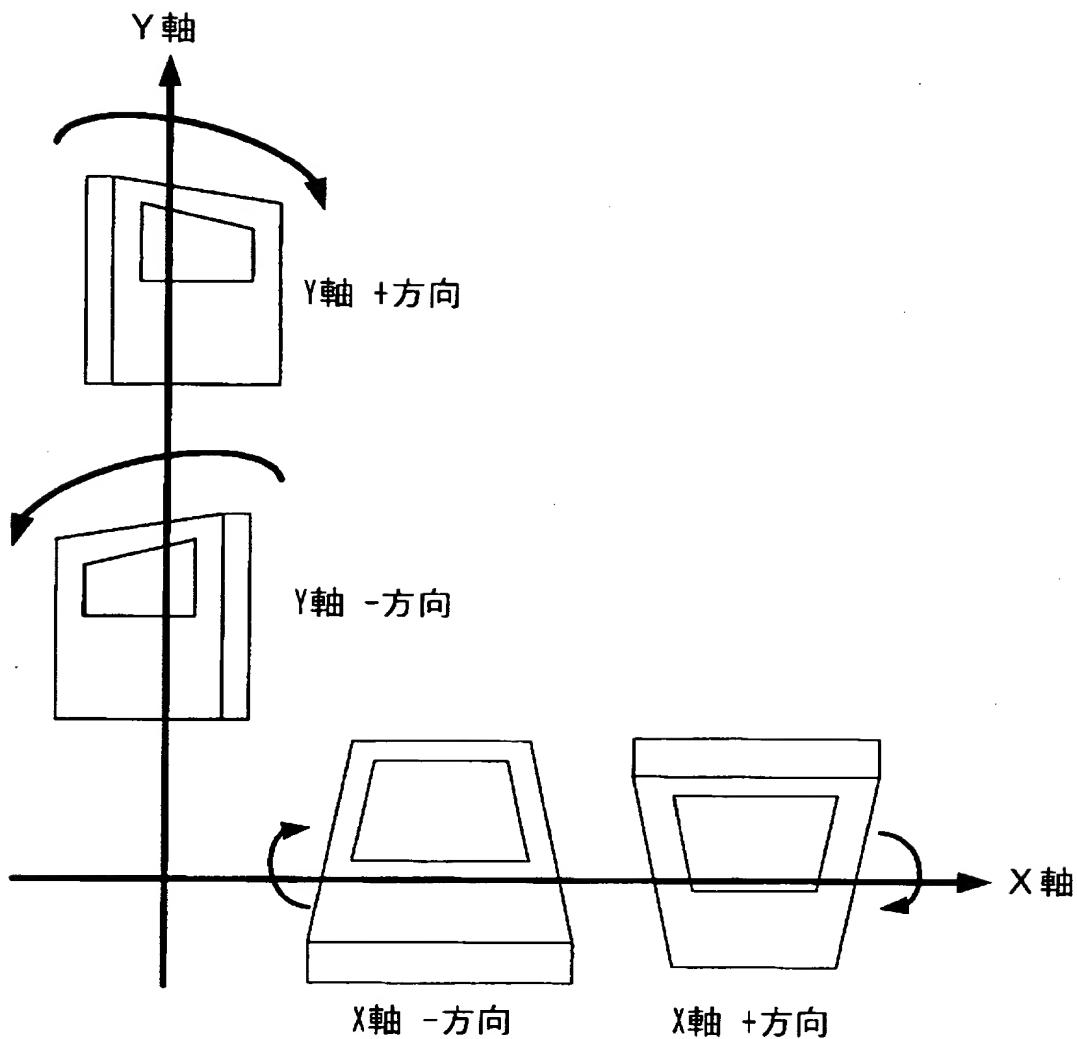
【図8】



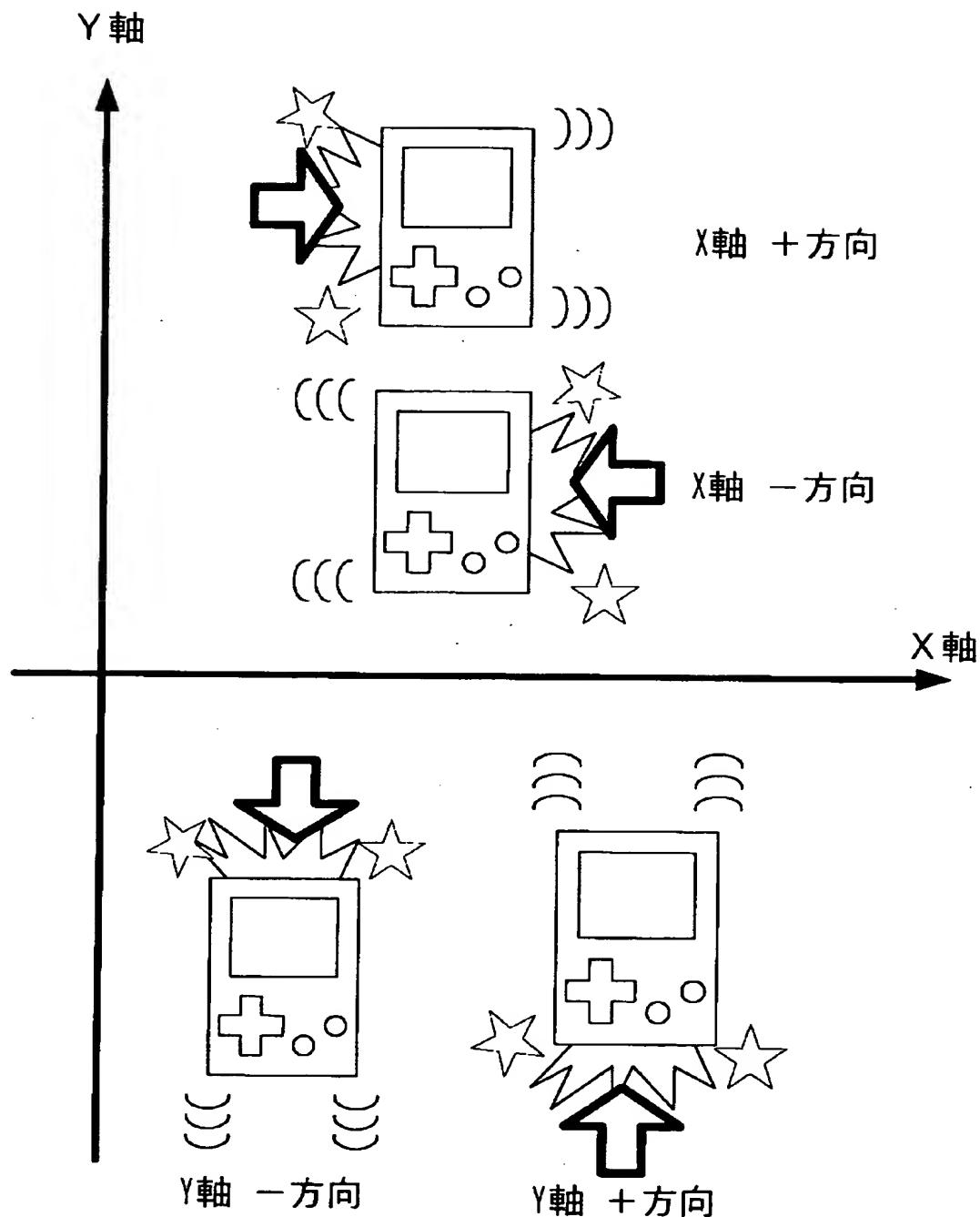
【図9】



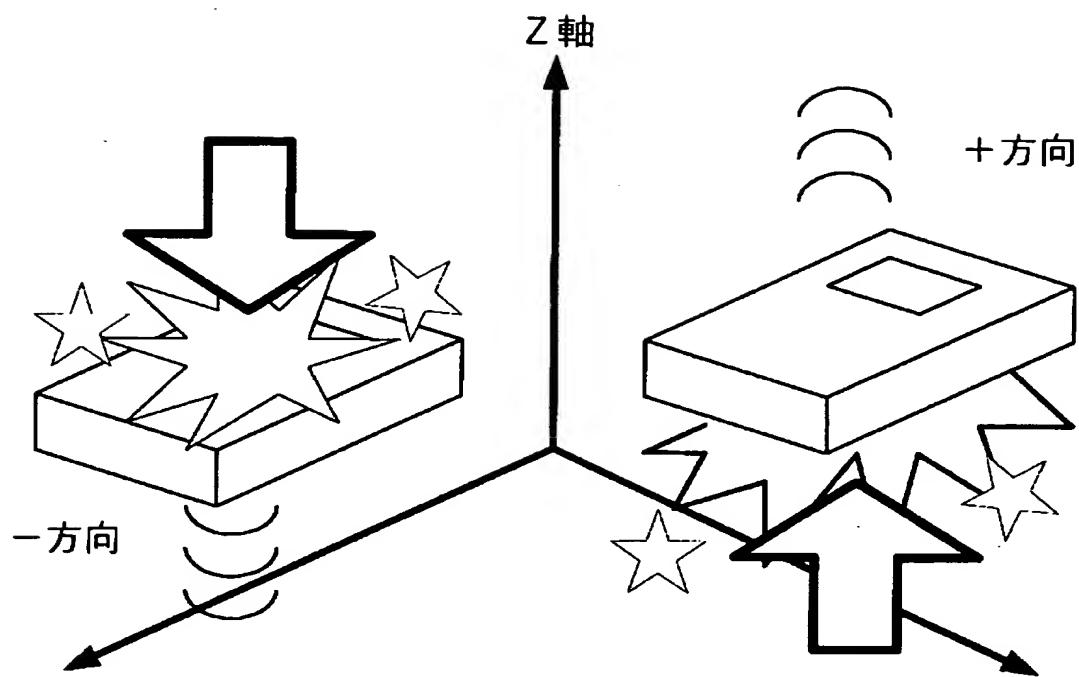
【図10】



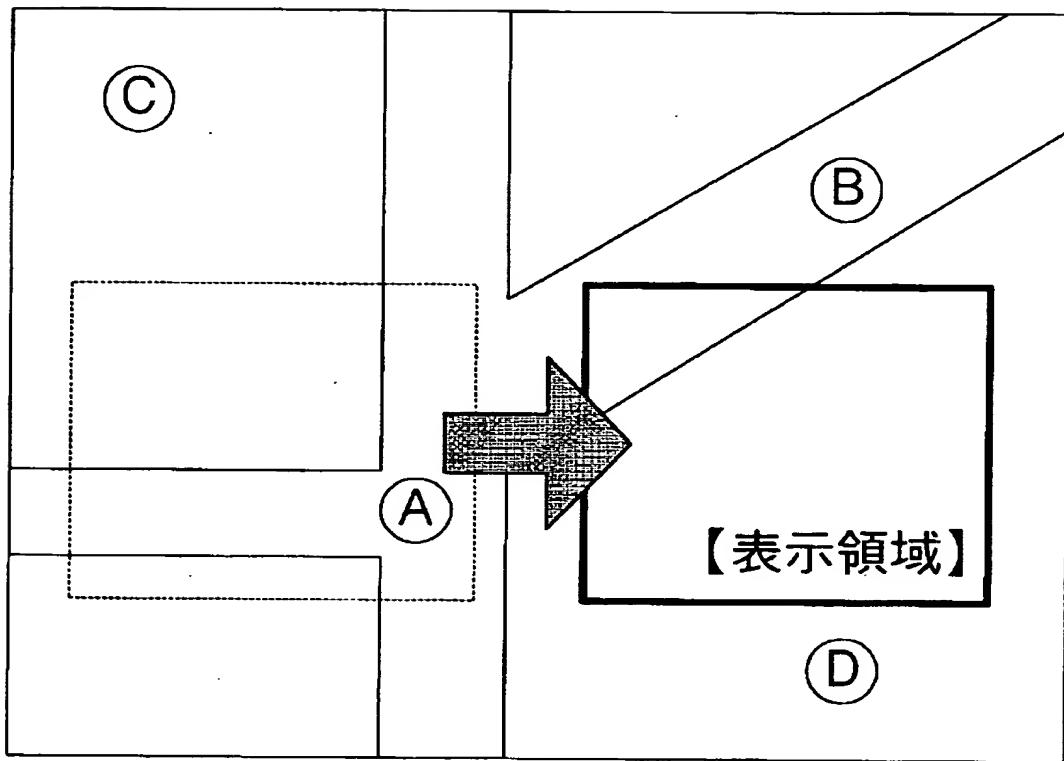
【図11】



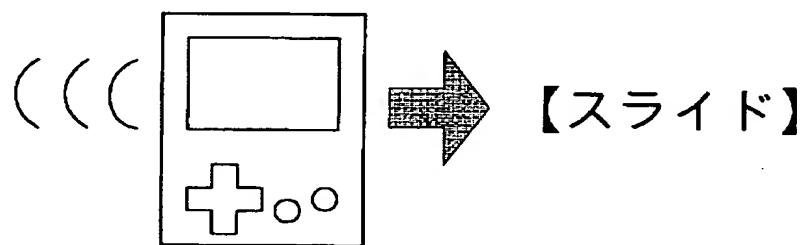
【図12】



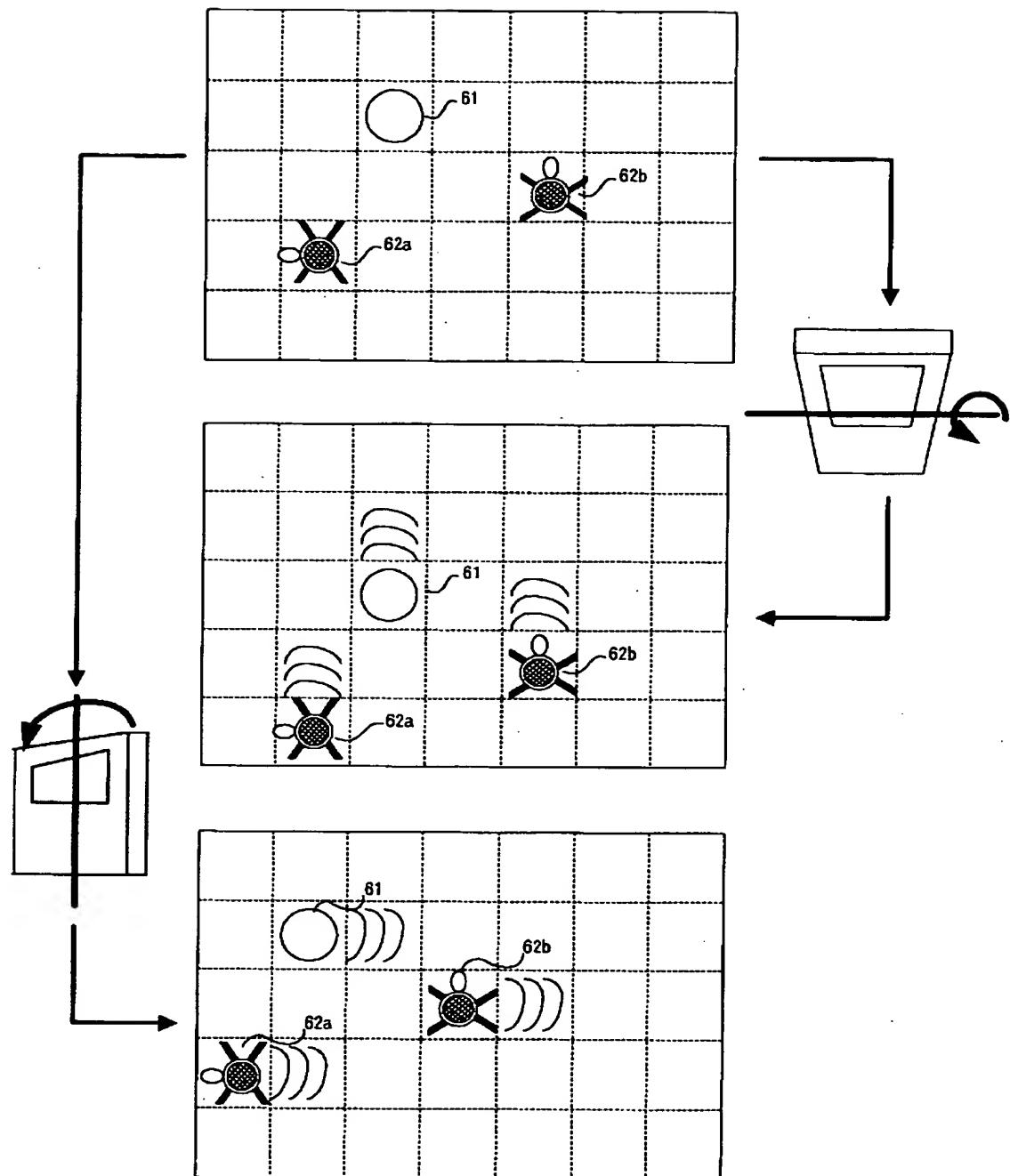
【図13】



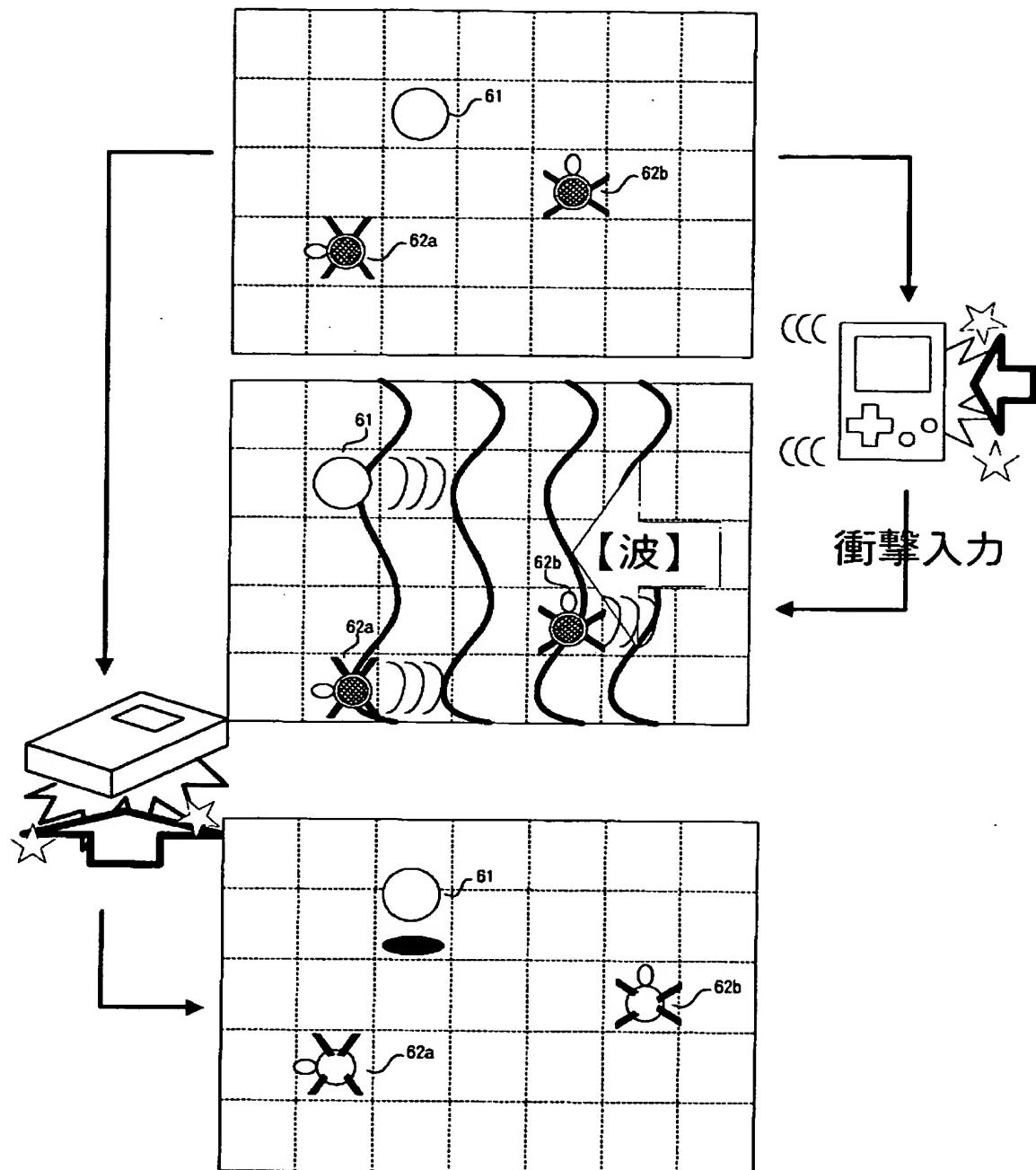
【仮想マップ】



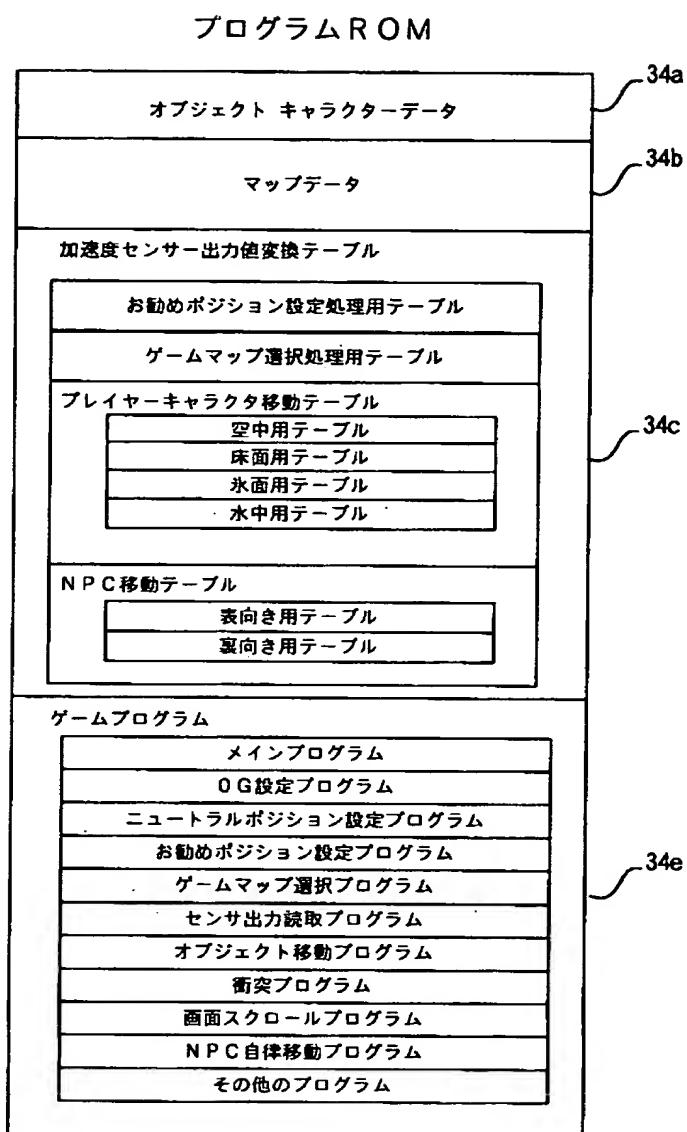
【図14】



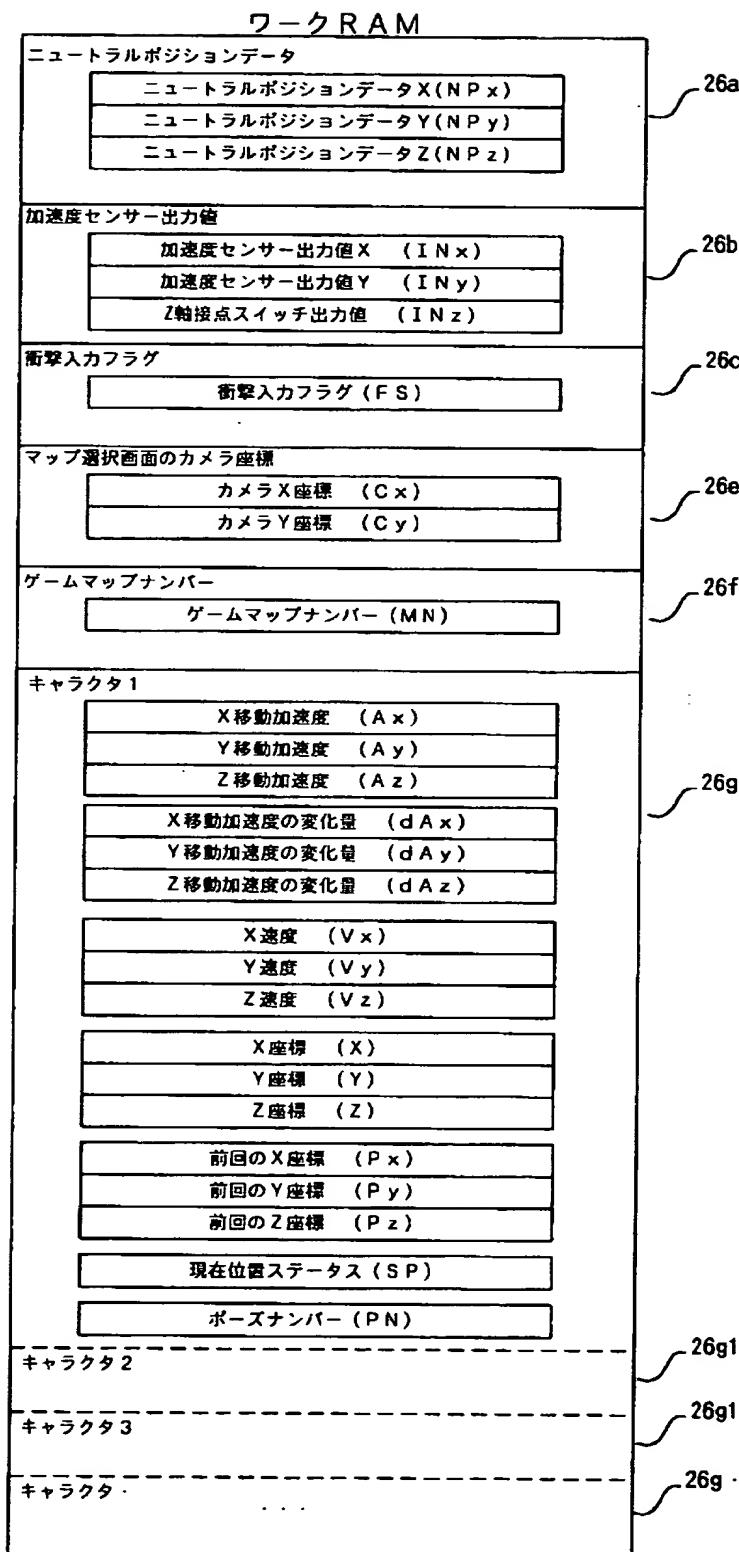
【図15】



【図16】

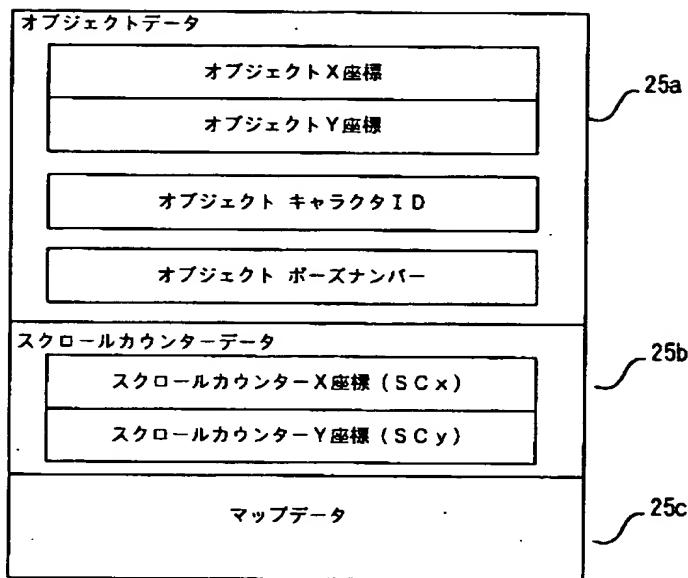


【図17】



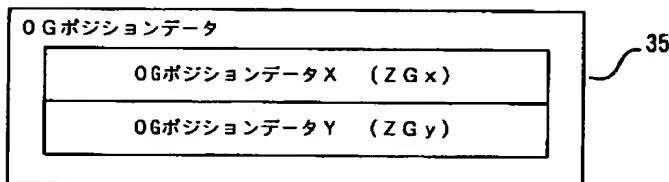
【図18】

表示用RAM



【図19】

バックアップRAM



【図20】

## ゲームマップ選択処理用テーブル

	利用方法	補正比率	特殊補正条件1	特殊補正数1	特殊補正条件2	特殊補正数2
センサ出力値X (INx)	カメラX座標 (Cx)の変化量	×2	—	—	—	—
センサ出力値X (INx)	カメラY座標 (Cy)の変化量	×2	—	—	—	—
Z軸接点スイッチ出力値(INz)	マップ決定	—	—	—	—	—
衝撃入力フラグ (FS)	—	—	—	—	—	—

【図21】

## プレイヤーキャラクタ移動用テーブル（空中用）

	利用方法	補正比率	特殊補正条件1	特殊補正数1	特殊補正条件2	特殊補正数2
センサ出力値X (INx)	—	—	—	—	—	—
センサ出力値X (INx)	—	—	—	—	—	—
Z軸接点スイッチ出力値(INz)	Z移動加速度の 変化量(dAz)	×1	—	—	—	—
衝撃入力フラグ (FS)	—	—	—	—	—	—

【図22】

## プレイヤーキャラクタ移動用テーブル（床面用）

	利用方法	補正比率	特殊補正条件1	特殊補正数1	特殊補正条件2	特殊補正数2
センサ出力値X (INx)	X移動加速度の 変化量(dAx)	×2	INx>20	40	—	—
センサ出力値X (INx)	Y移動加速度の 変化量(dAy)	×2	INx>20	40	—	—
Z軸接点スイッチ出力値(INz)	Z移動加速度の 変化量(dAz)	×1	—	—	—	—
衝撃入力フラグ (FS)	X,Y移動加速度 の変化量 (dAx, dAy)	×3	—	—	—	—

【図23】

## プレイヤキャラクタ移動用テーブル（水面用）

	利用方法	補正比率	特殊補正条件1	特殊補正数1	特殊補正条件2	特殊補正数2
センサ出力値X (INx)	X移動加速度の 変化量(dAx)	×3	INx>20	60	—	—
センサ出力値X (INx)	Y移動加速度の 変化量(dAy)	×3	INy>20	60	—	—
Z軸接点スイッ チ出力値(INz)	Z移動加速度の 変化量(dAz)	×1	—	—	—	—
衝撃入力フラグ (FS)	X,Y移動加速度 の変化量 (dAx,dAy)	×5	—	—	—	—

【図24】

## プレイヤキャラクタ移動用テーブル（水中用）

	利用方法	補正比率	特殊補正条件1	特殊補正数1	特殊補正条件2	特殊補正数2
センサ出力値X (INx)	X移動加速度の 変化量(dAx)	×1/2	INx>10	5	—	—
センサ出力値X (INx)	Y移動加速度の 変化量(dAy)	×1/2	INy>10	5	—	—
Z軸接点スイッ チ出力値(INz)	Z移動加速度の 変化量(dAz)	×1	—	—	—	—
衝撃入力フラグ (FS)	—	—	—	—	—	—

【図25】

## NPC移動用テーブル（亀表向き用）

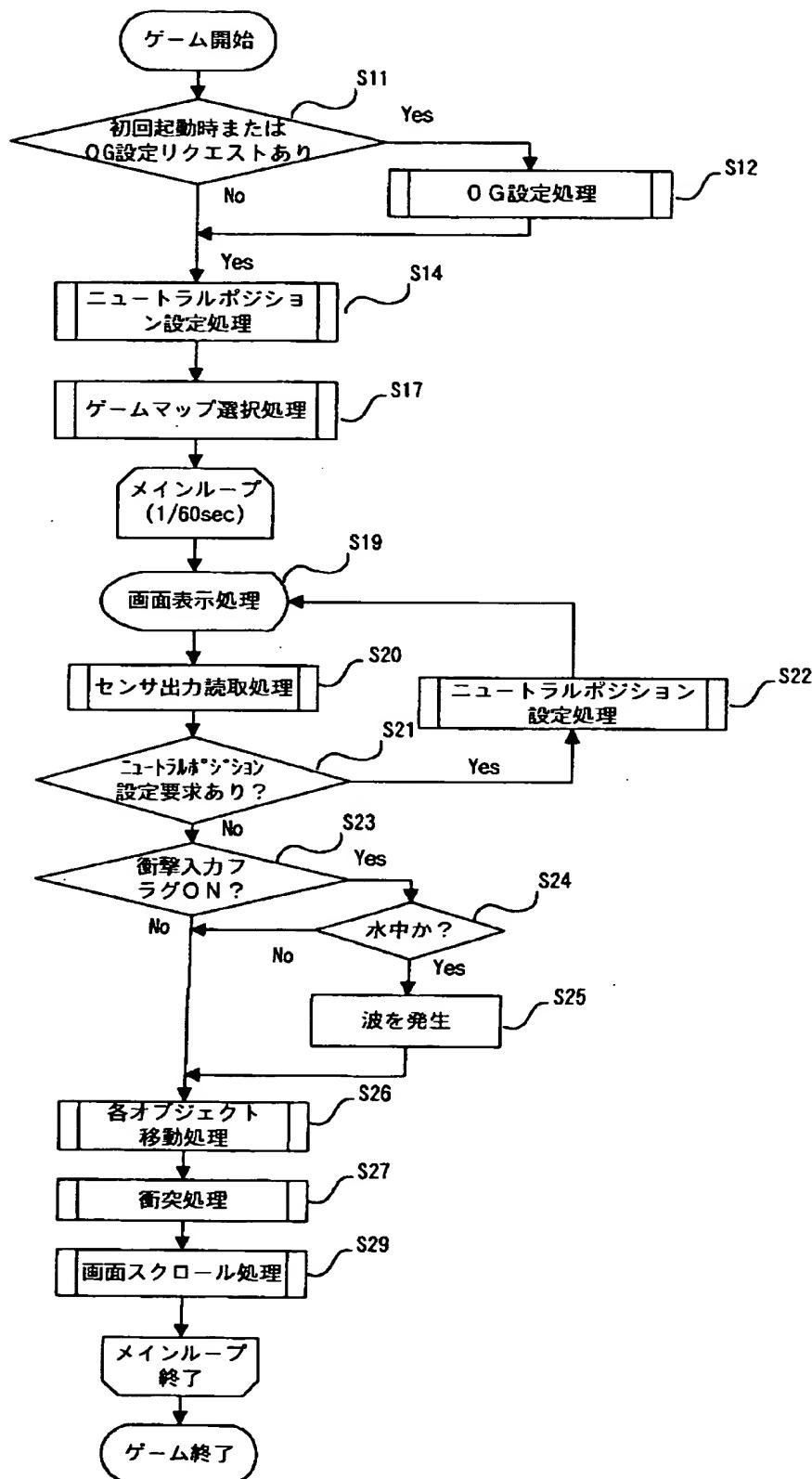
	利用方法	補正比率	特殊補正条件1	特殊補正数1	特殊補正条件2	特殊補正数2
センサ出力値X (INx)	X移動加速度の 変化量(dAx)	×1/2	INx<10	0	INx>20	10
センサ出力値X (INx)	Y移動加速度の 変化量(dAy)	×1/2	INy<10	0	INx>20	10
Z軸接点スイッ チ出力値(INz)	表裏逆転	—	—	—	—	—
衝撃入力フラグ (FS)	—	—	—	—	—	—

## 【図26】

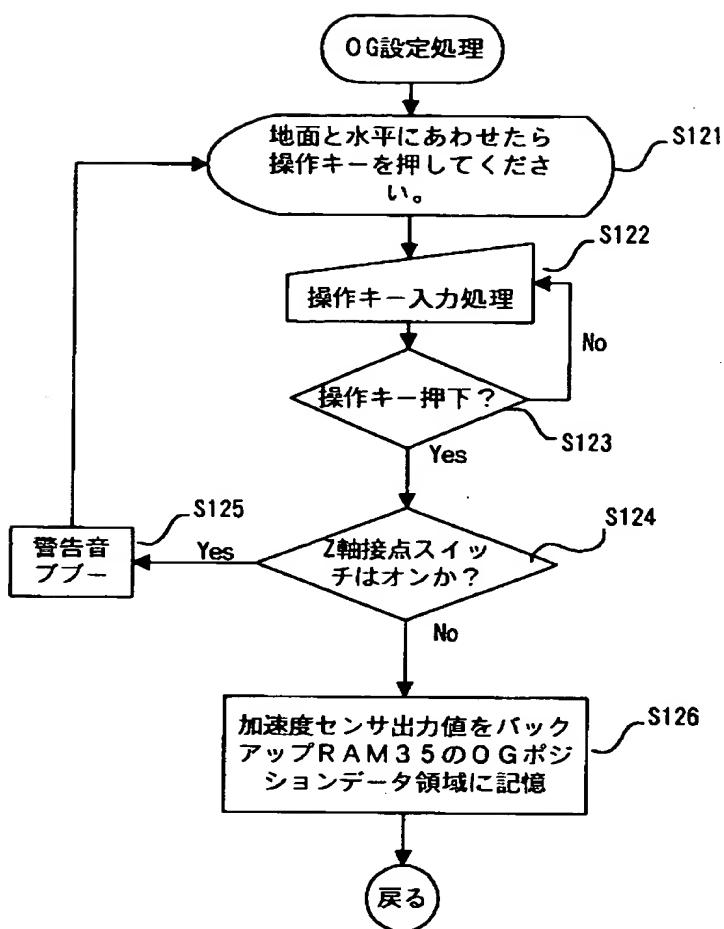
N P C移動用テーブル（亀裏向き用）

	利用方法	補正比率	特殊補正条件1	特殊補正数1	特殊補正条件2	特殊補正数2
センサ出力値X (INx)	X移動加速度の 変化量(dAx)	×2	INx>20	40	—	—
センサ出力値Y (INy)	Y移動加速度の 変化量(dAy)	×2	INy>20	40	—	—
Z軸接点スイッ チ出力値(INz)	表裏逆転	—	—	—	—	—
衝撃入力フラグ (FS)	—	—	—	—	—	—

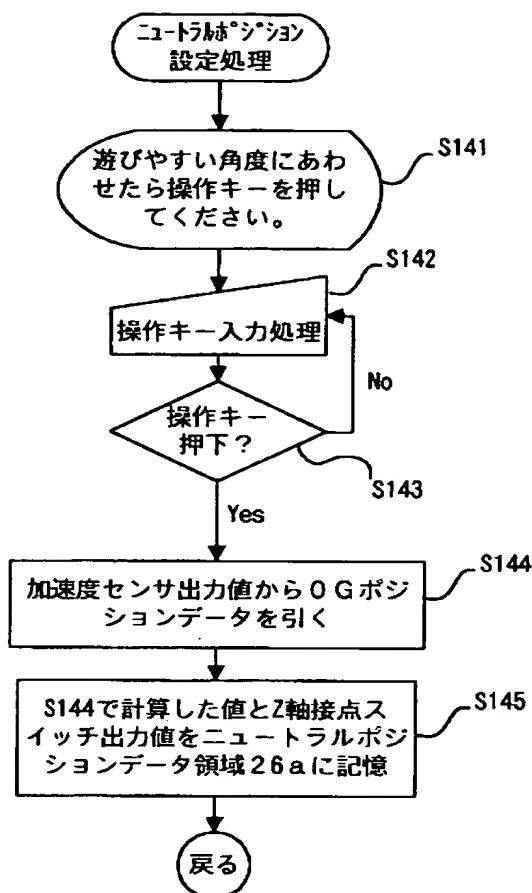
【図27】



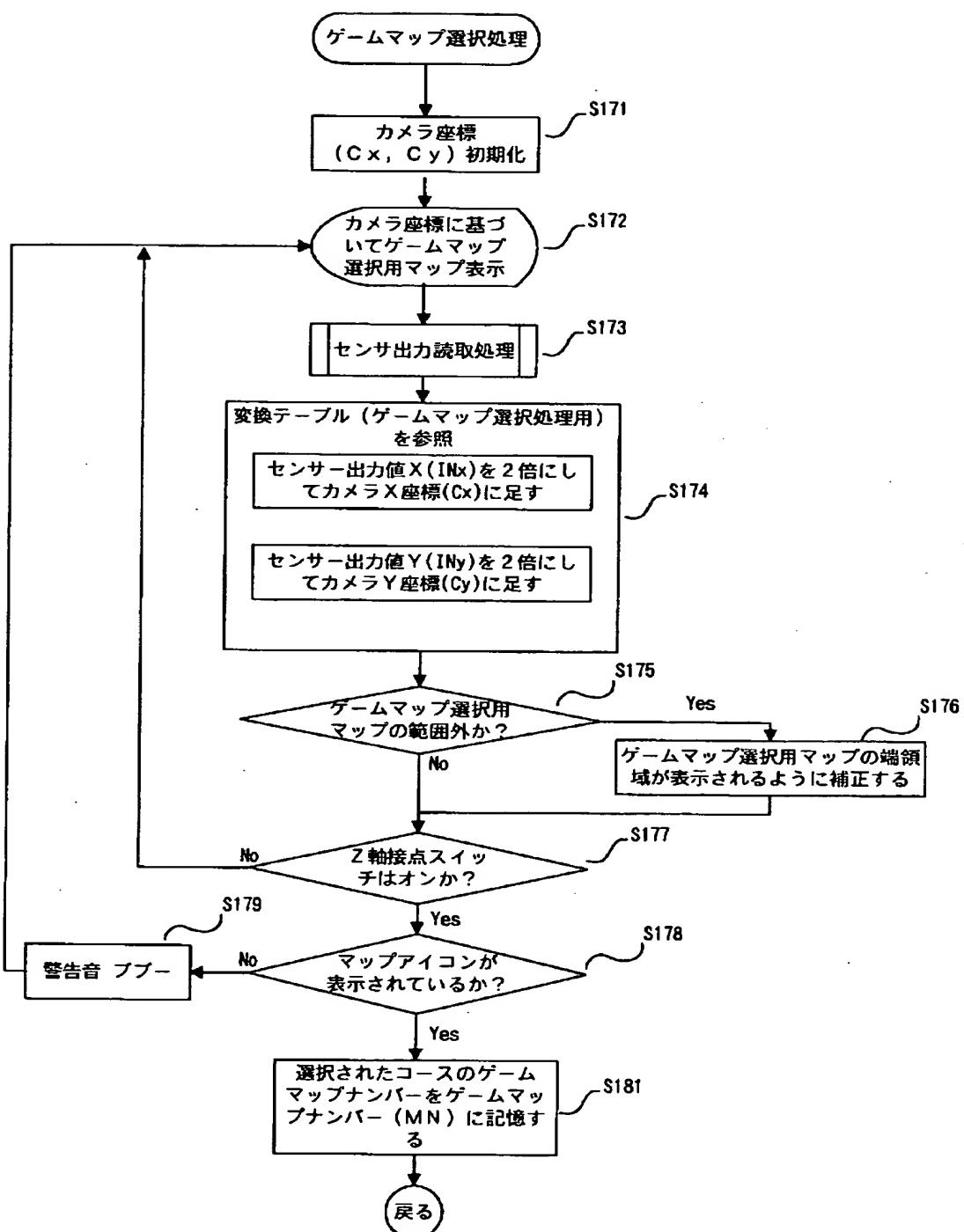
【図28】



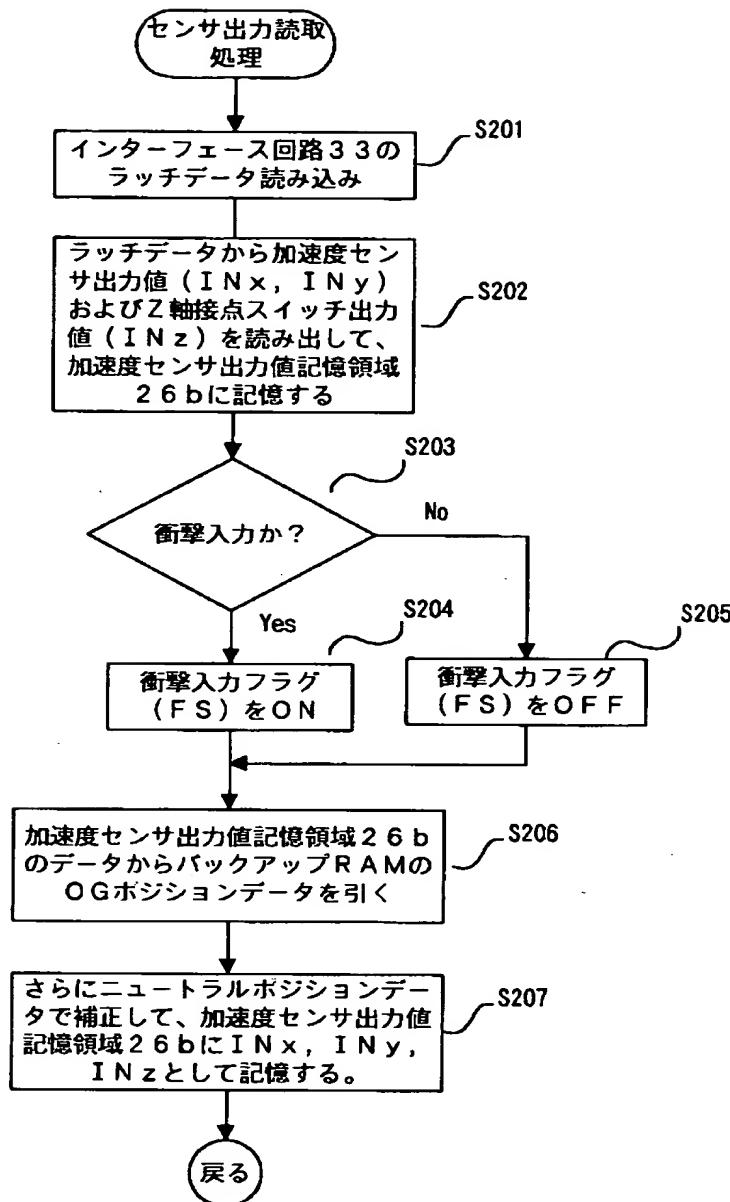
【図29】



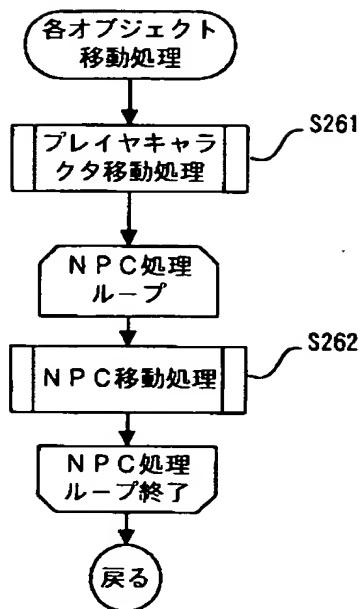
【図30】



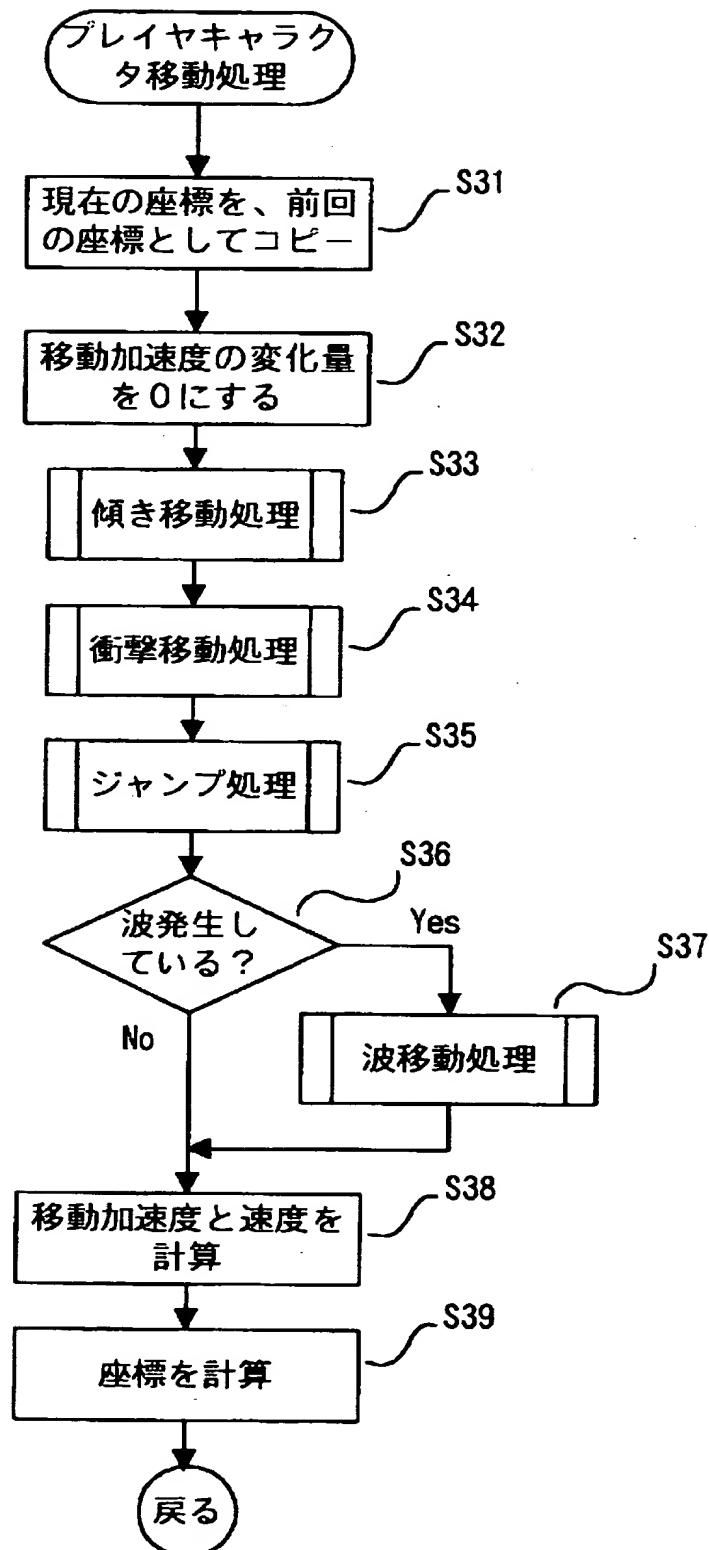
【図31】



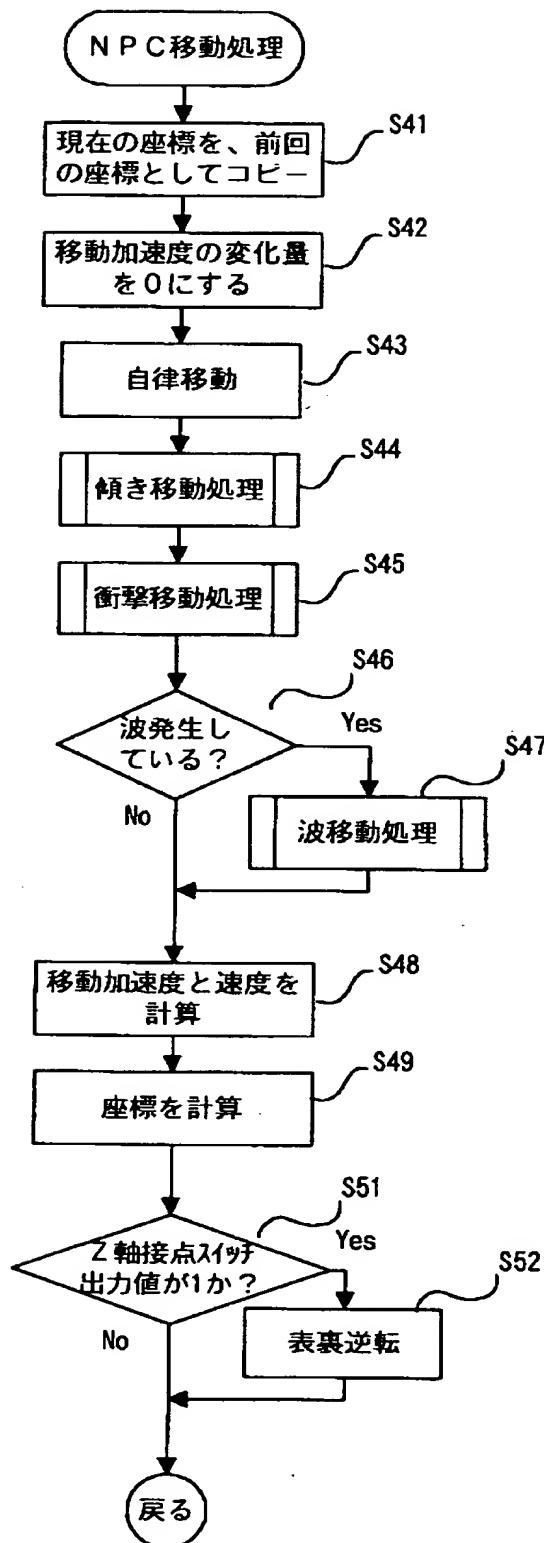
【図32】



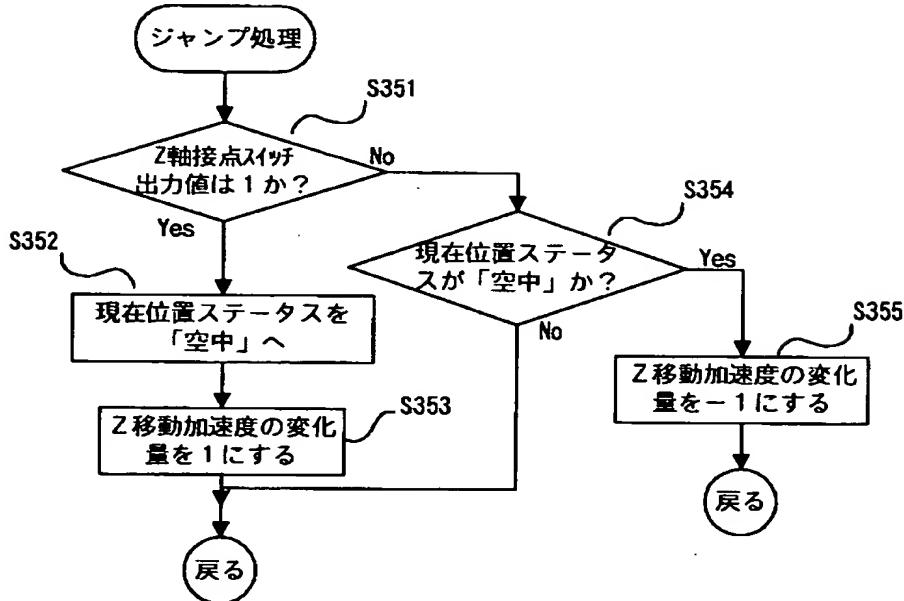
【図33】



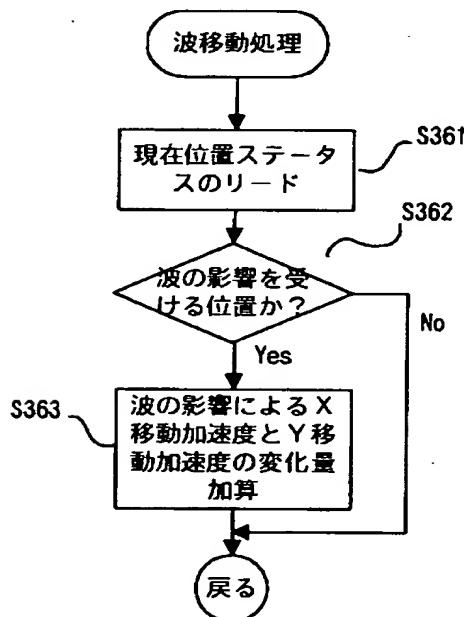
【図34】



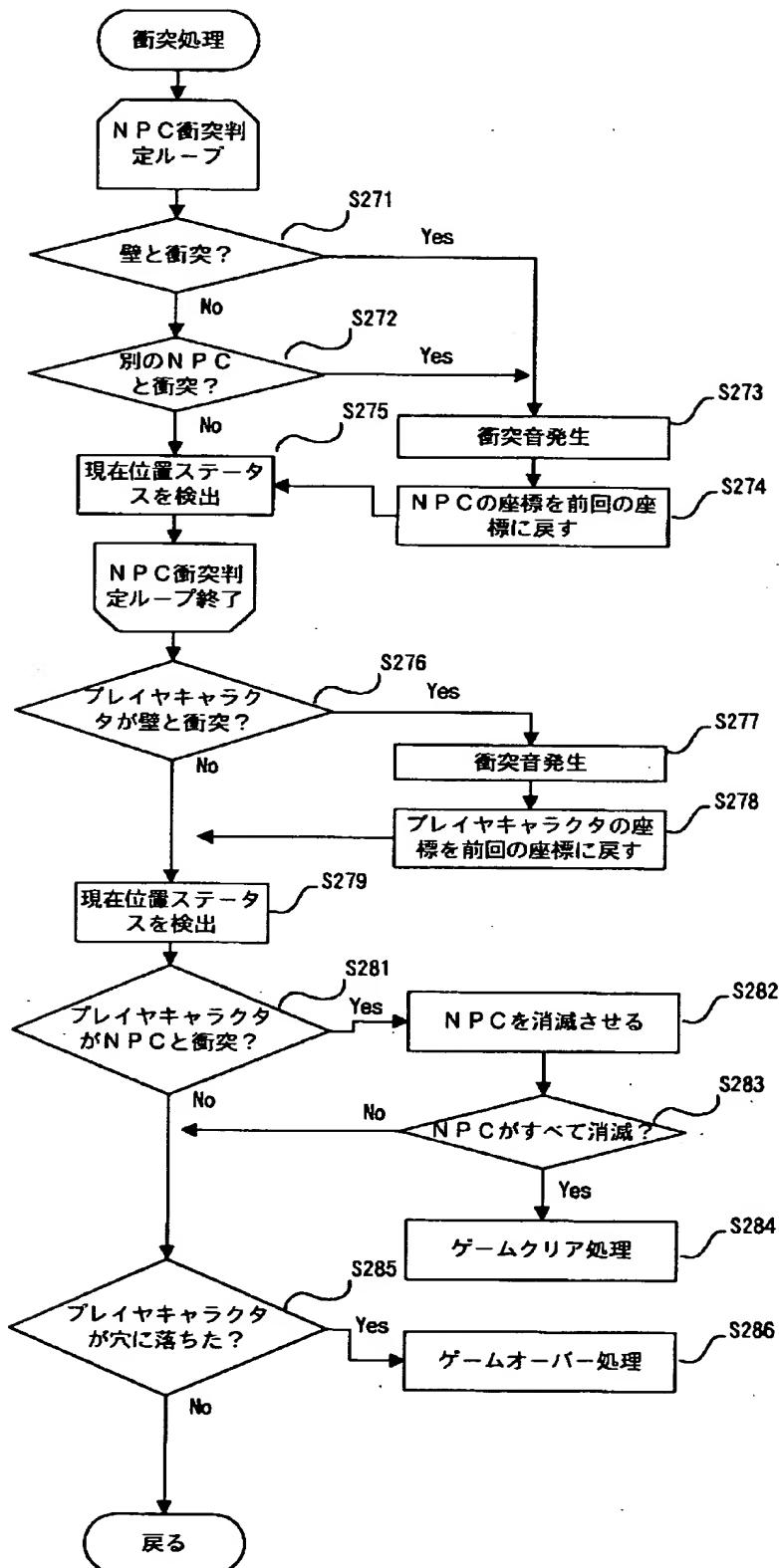
【図35】



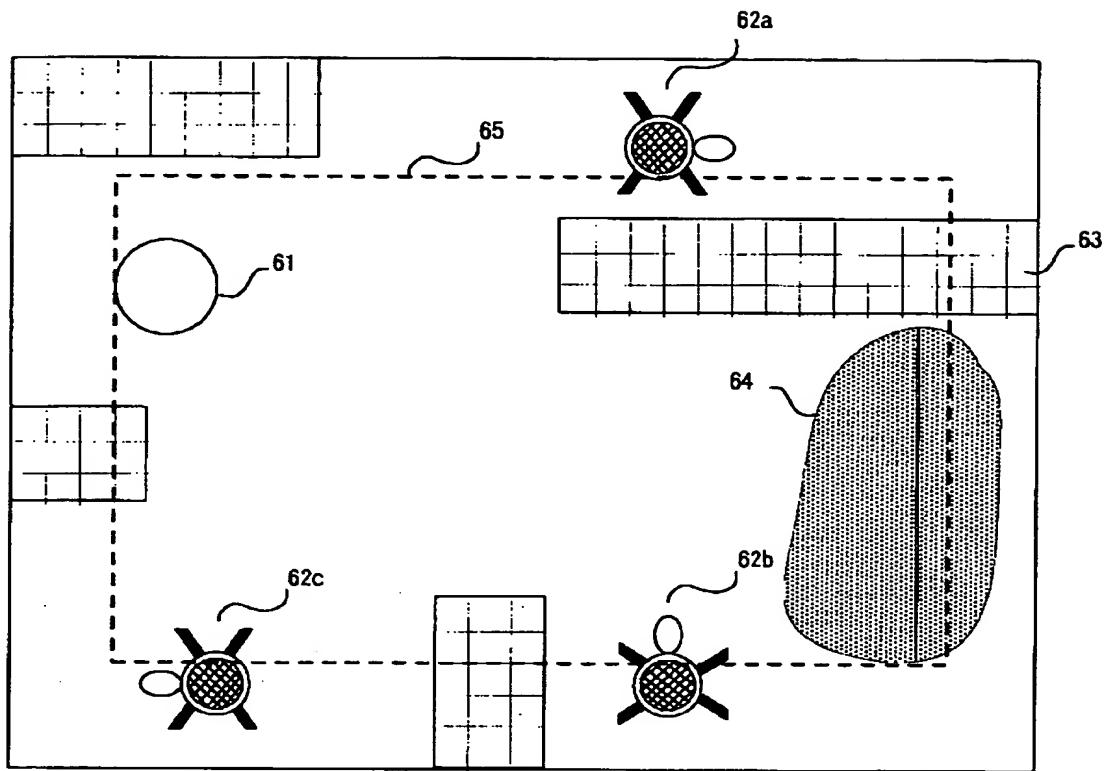
【図36】



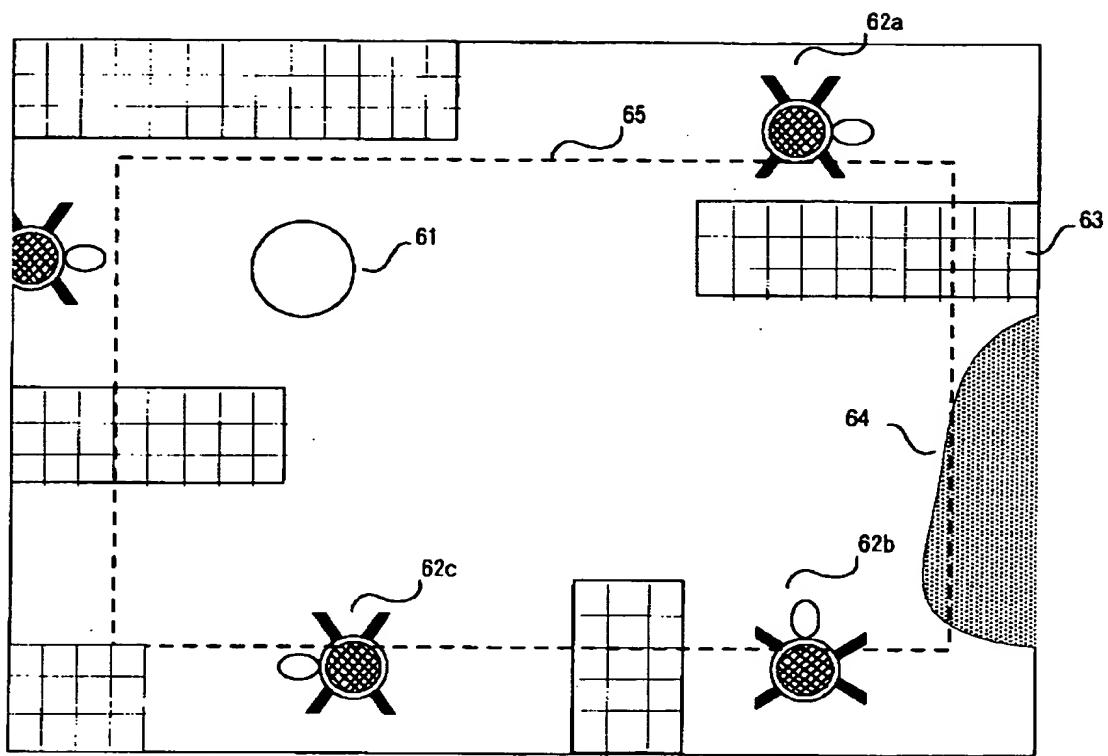
【図37】



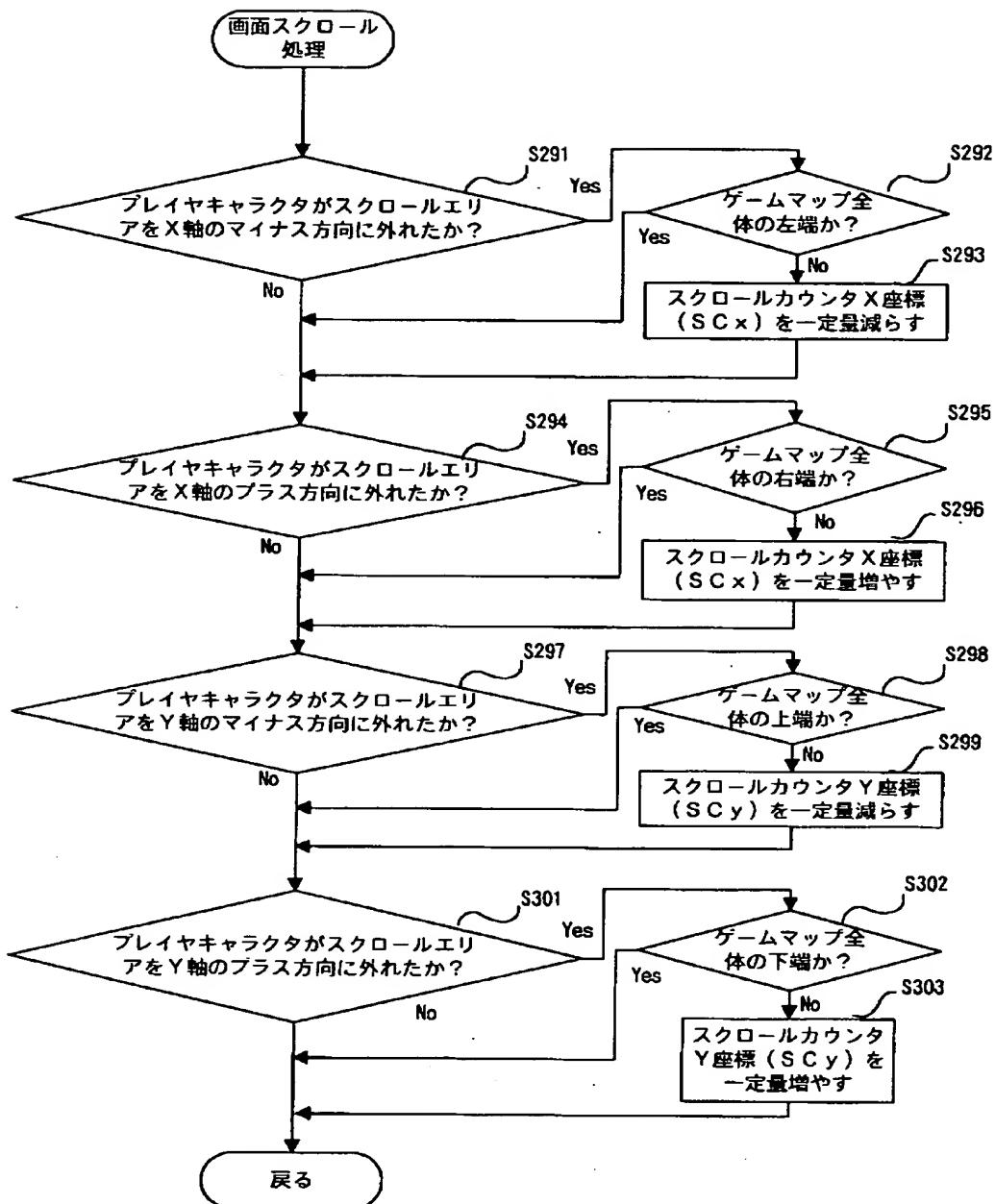
【図38】



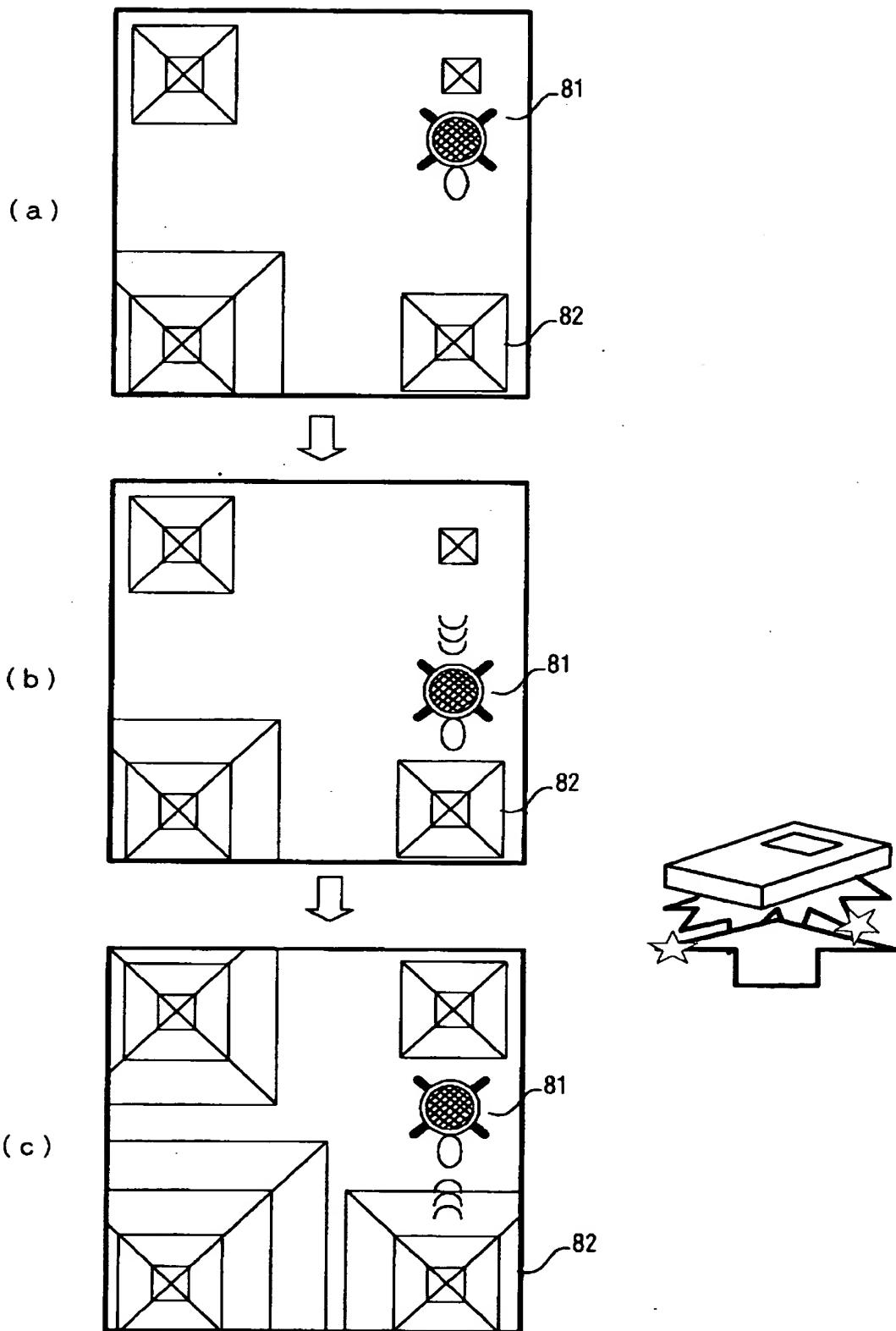
【図39】



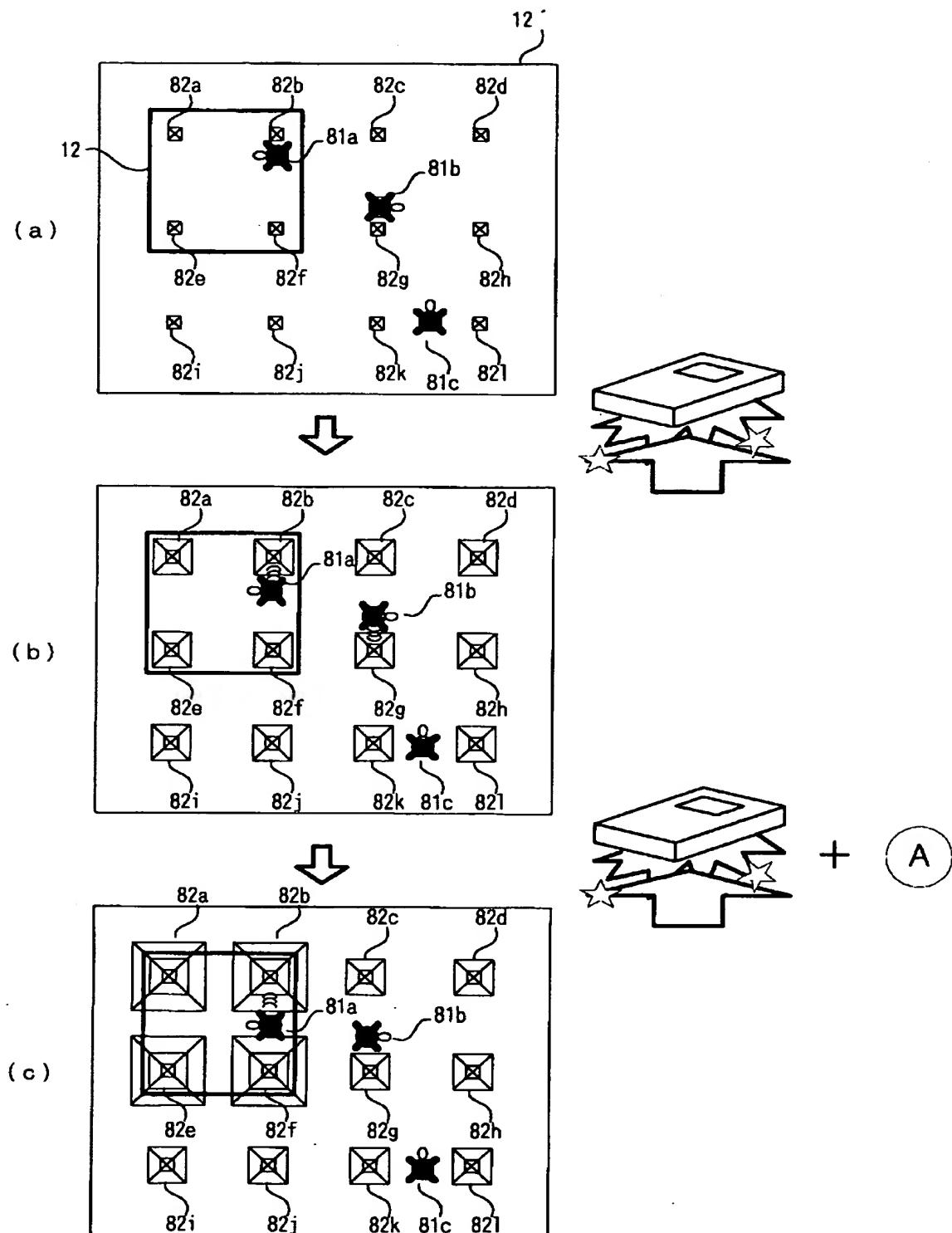
【図40】



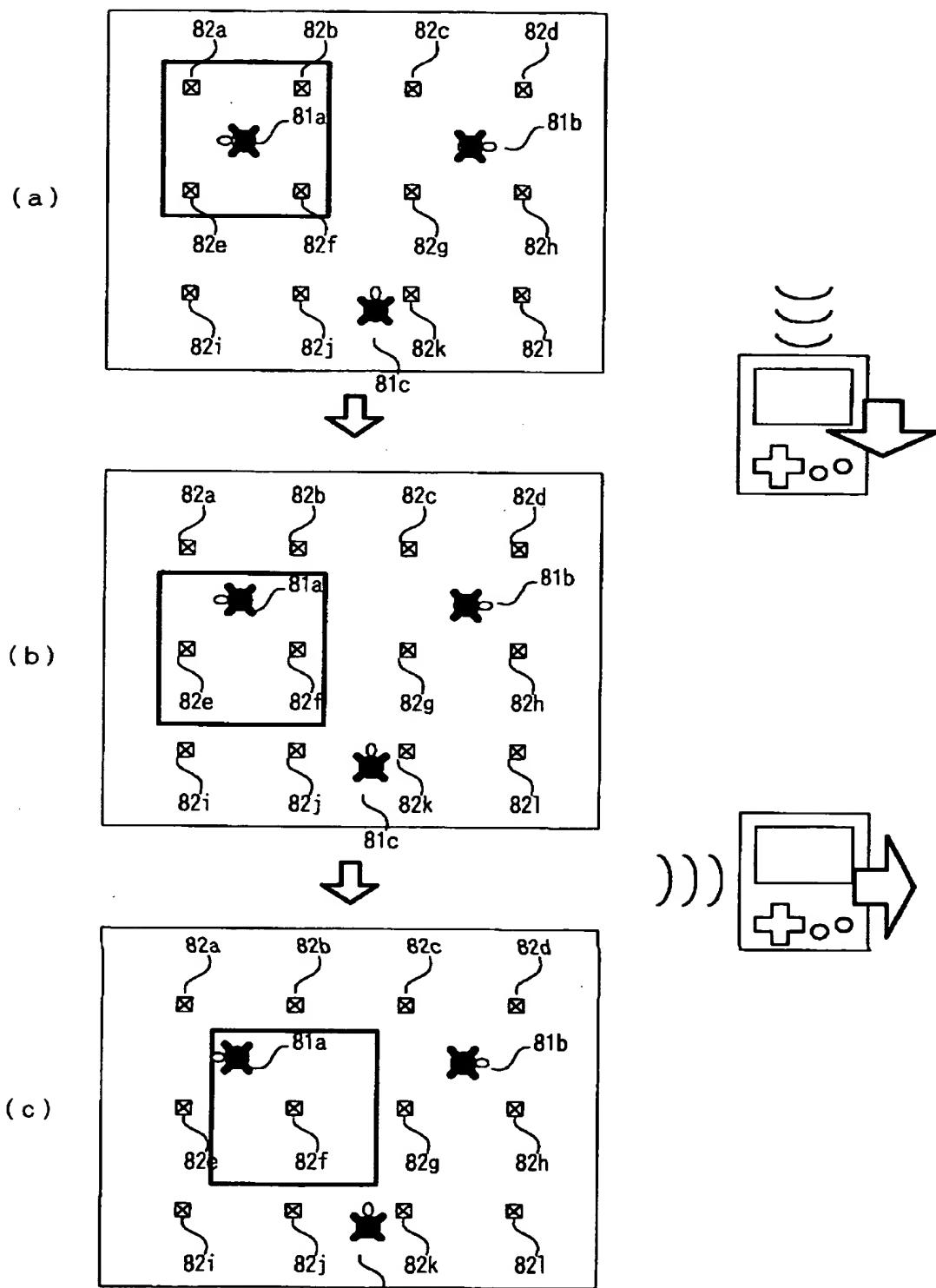
【図41】



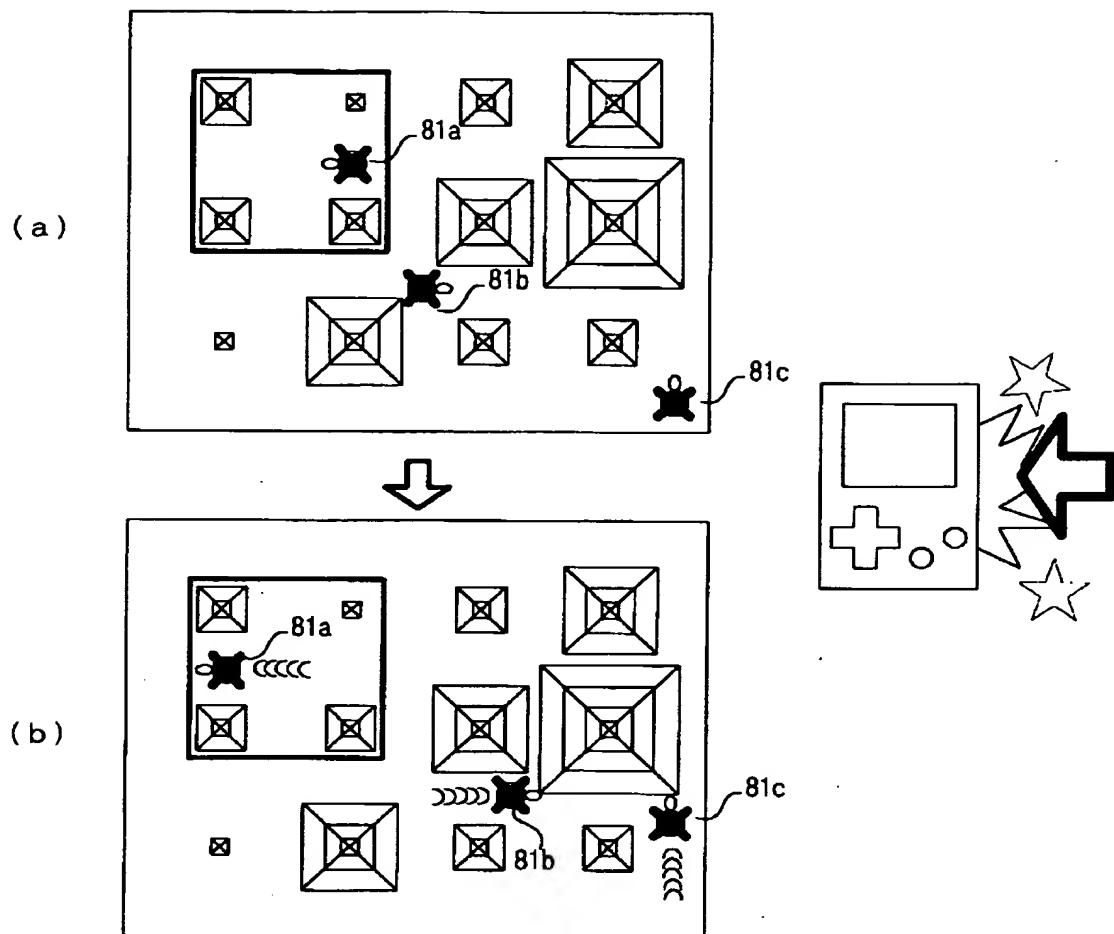
【図42】



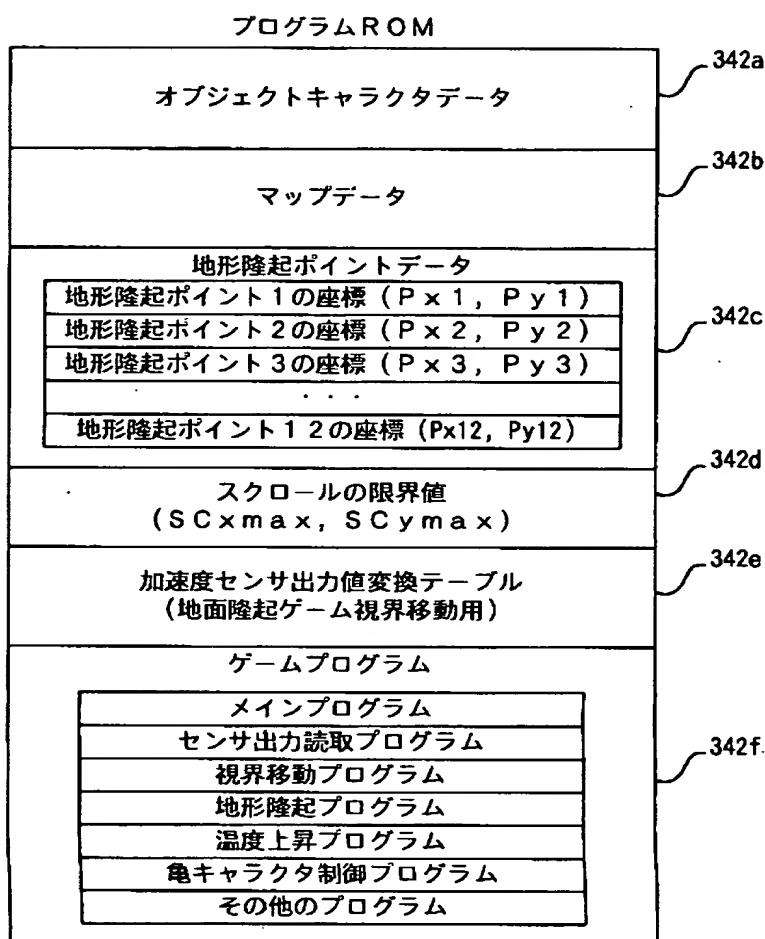
【図43】



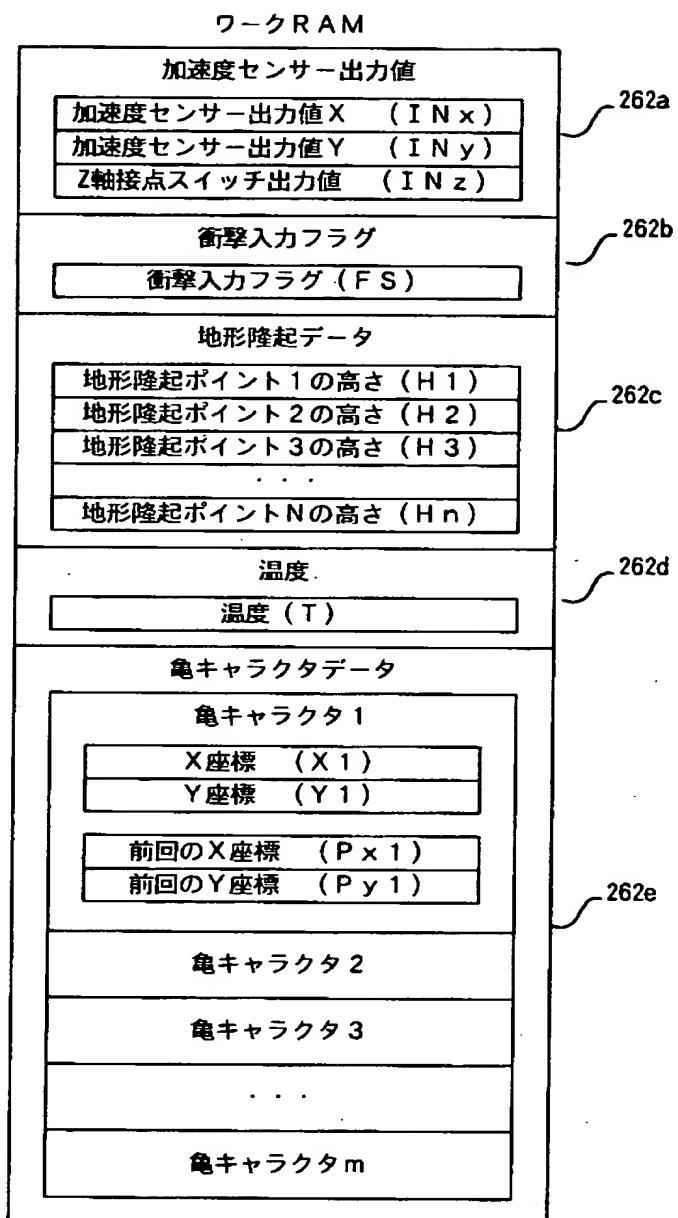
【図44】



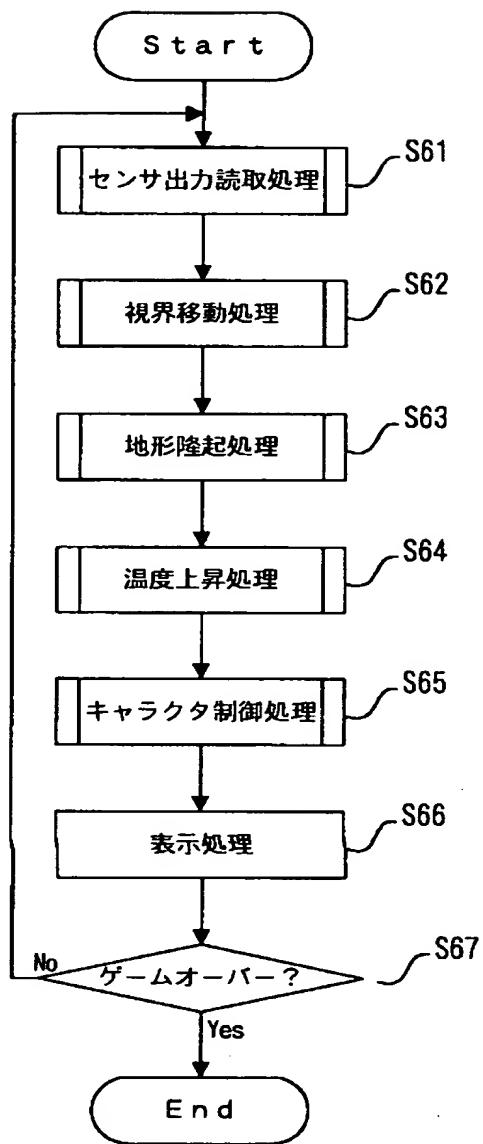
【図45】



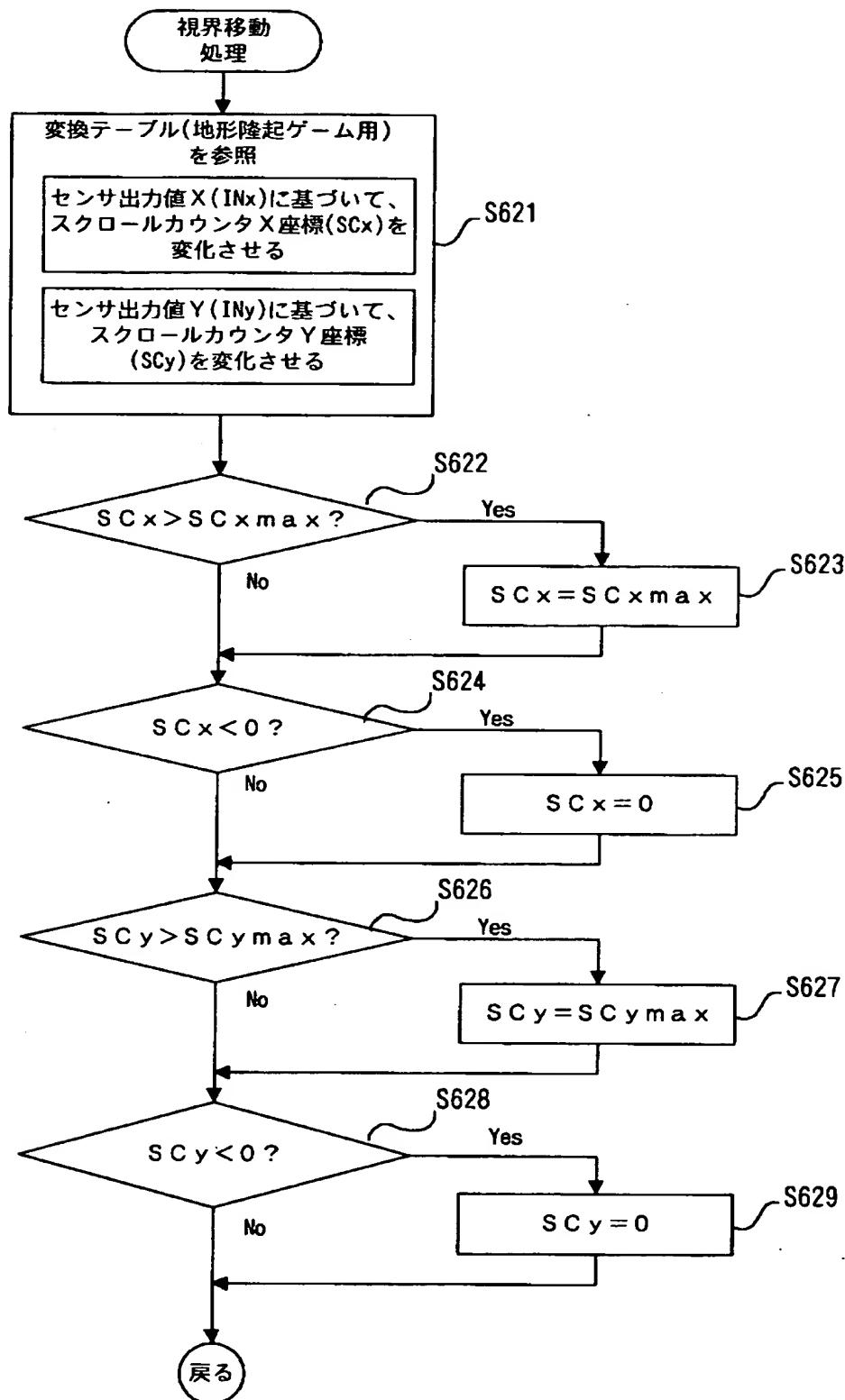
【図46】



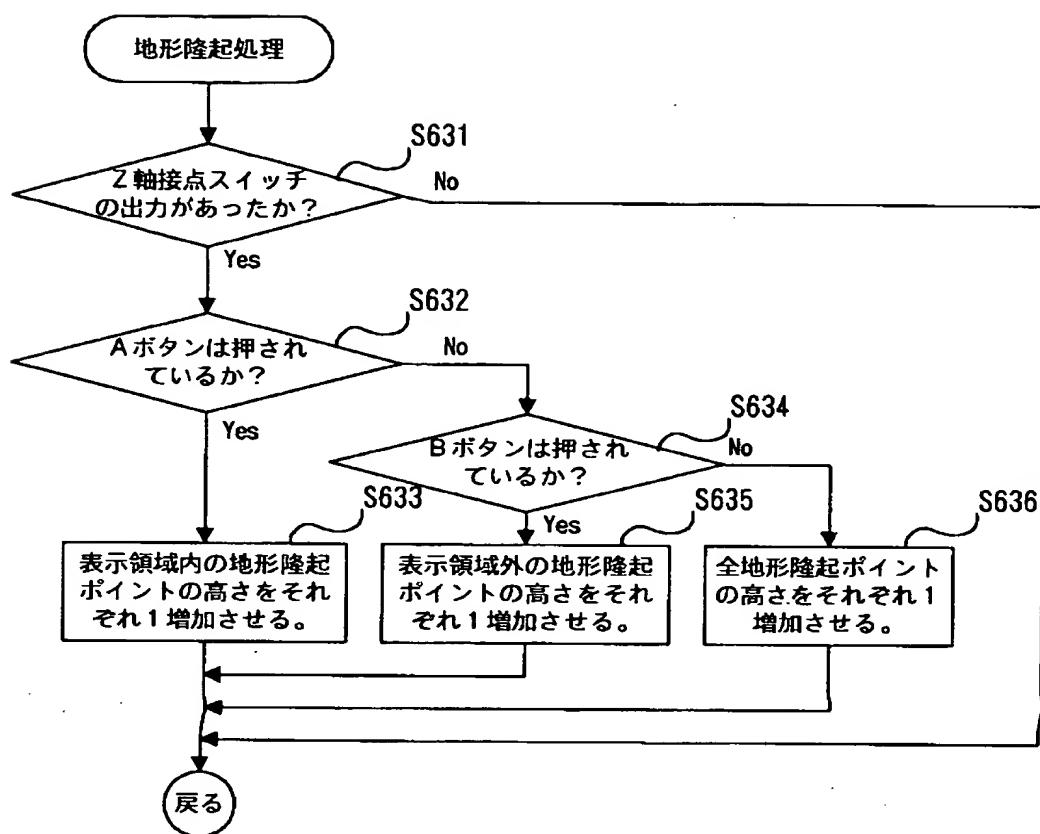
【図47】



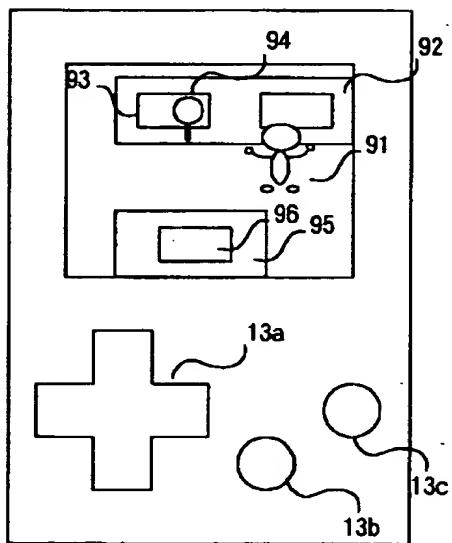
【図48】



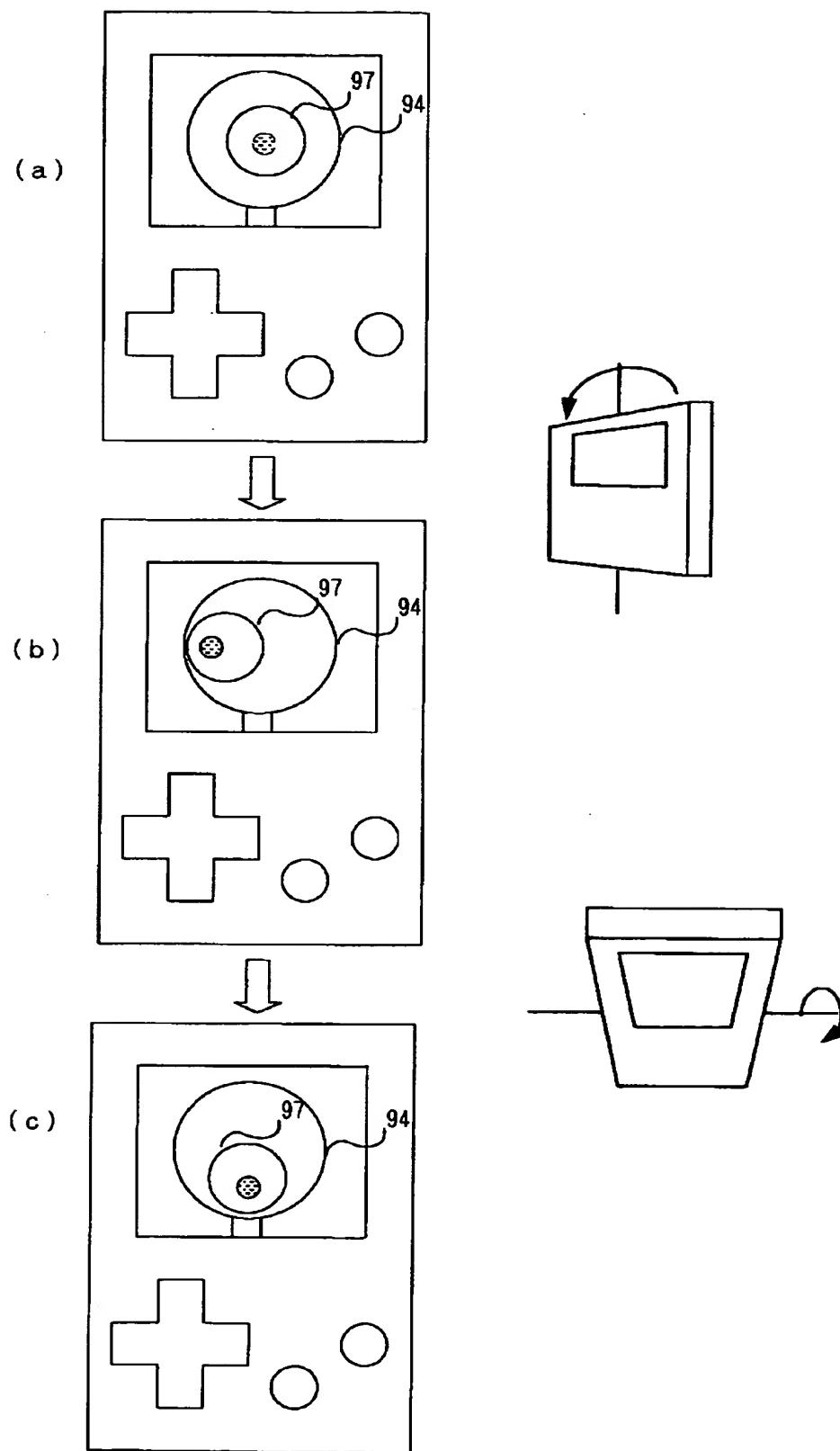
【図49】



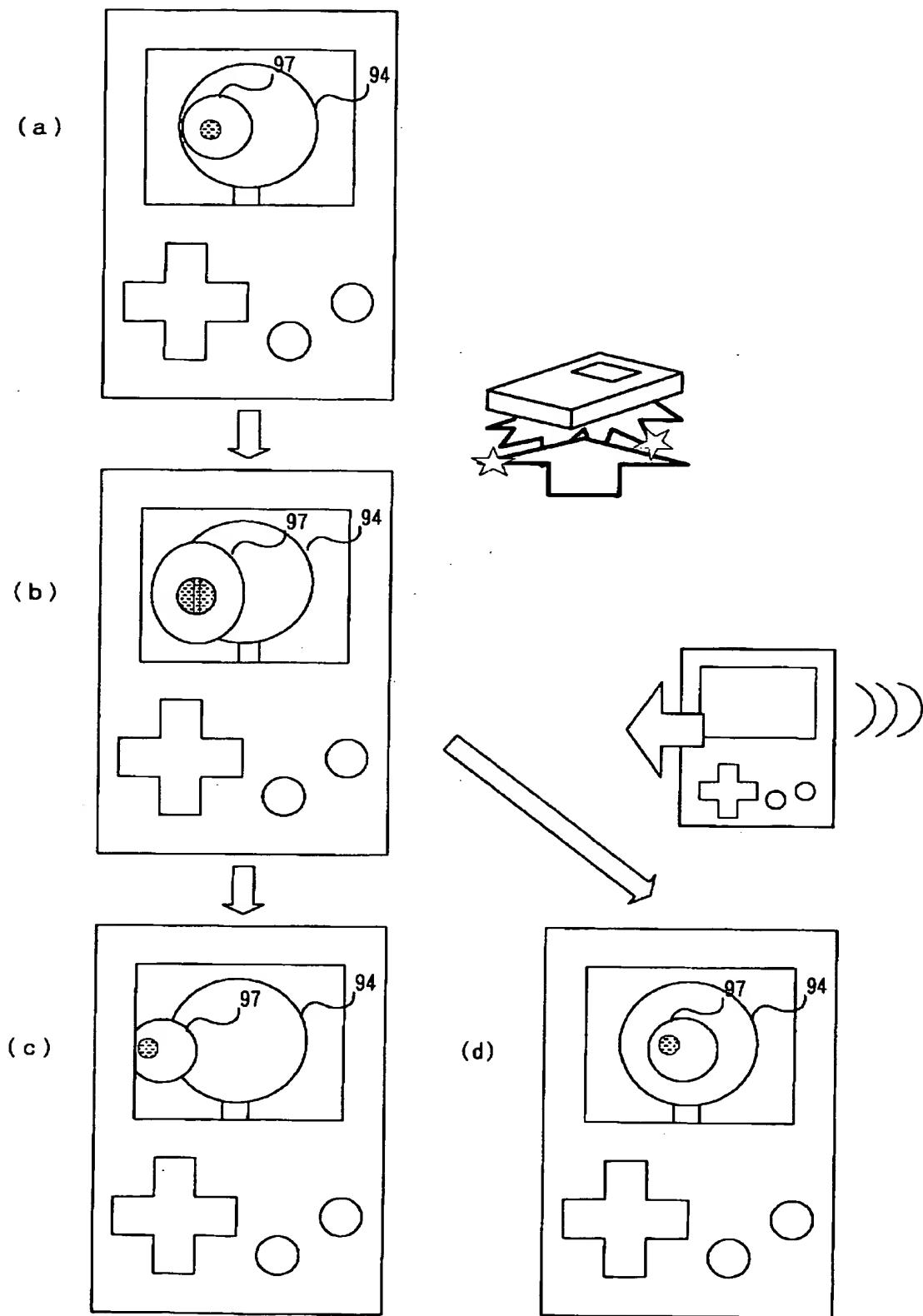
【図50】



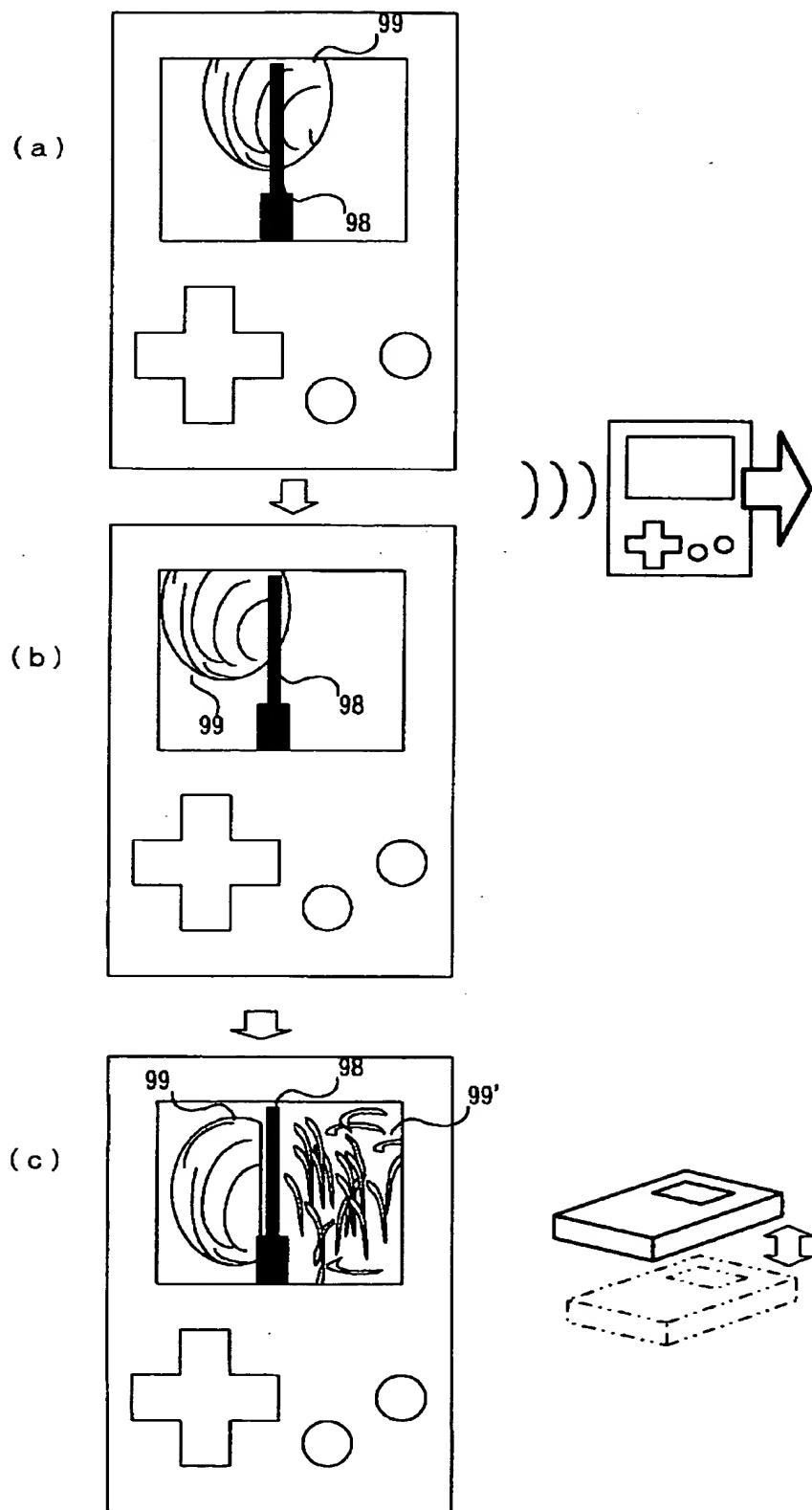
【図51】



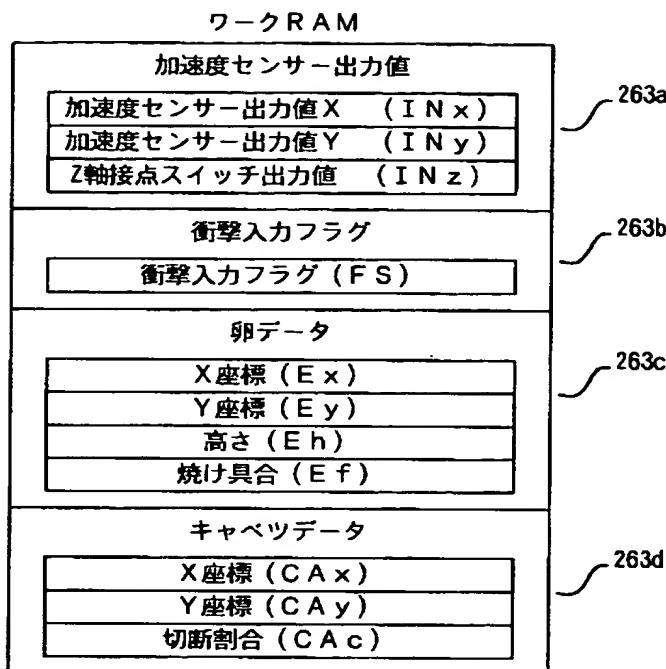
【図52】



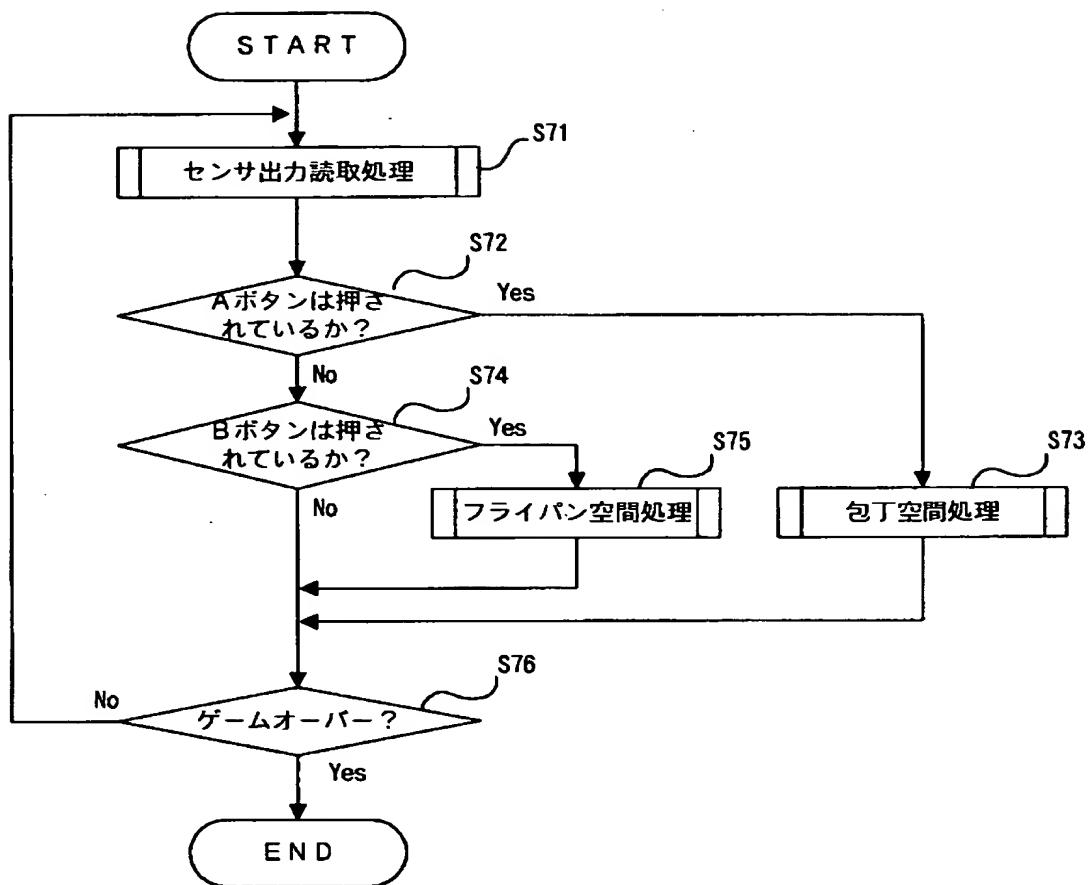
【図53】



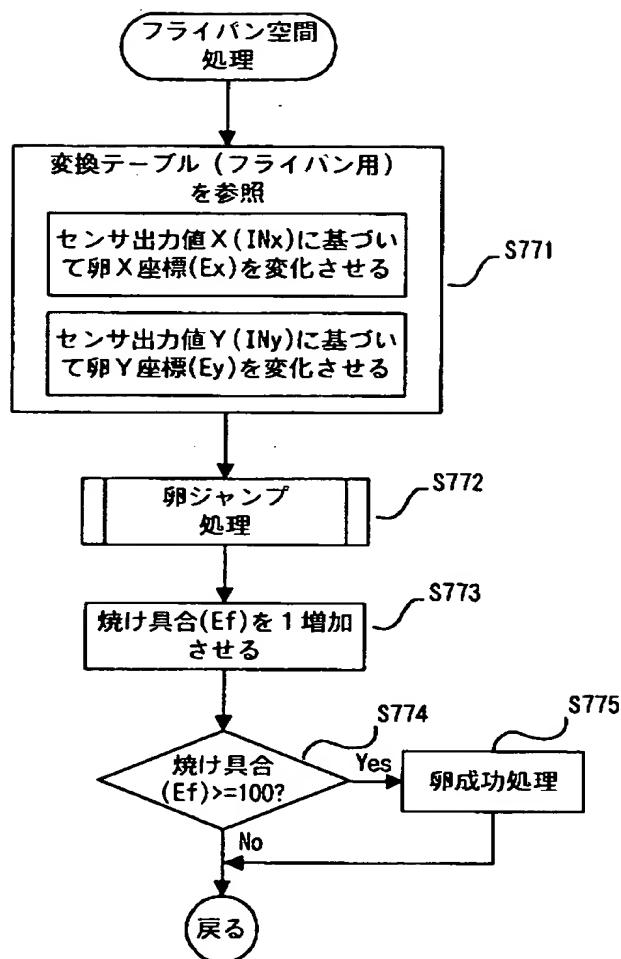
【図54】



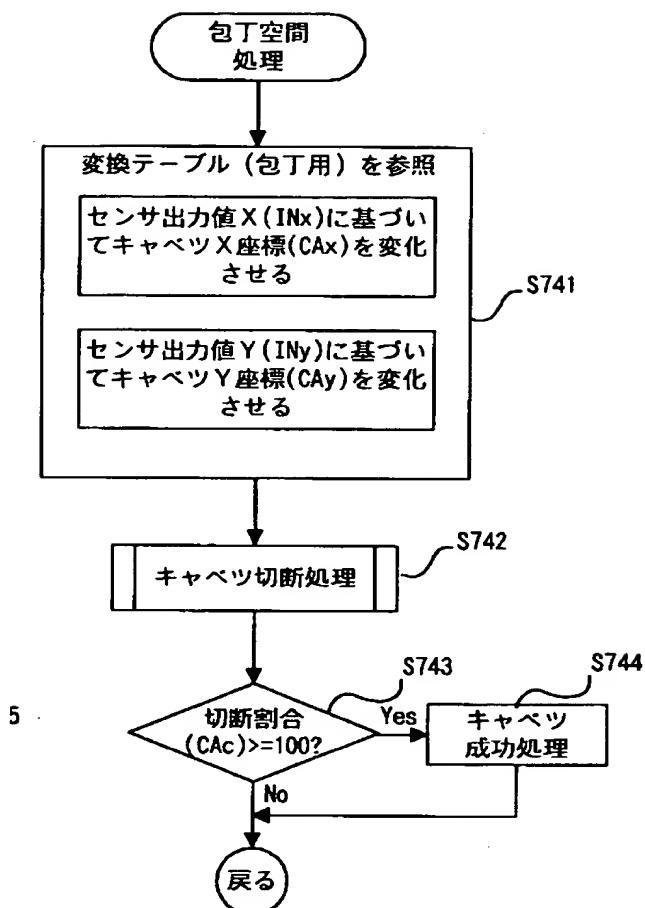
【図55】



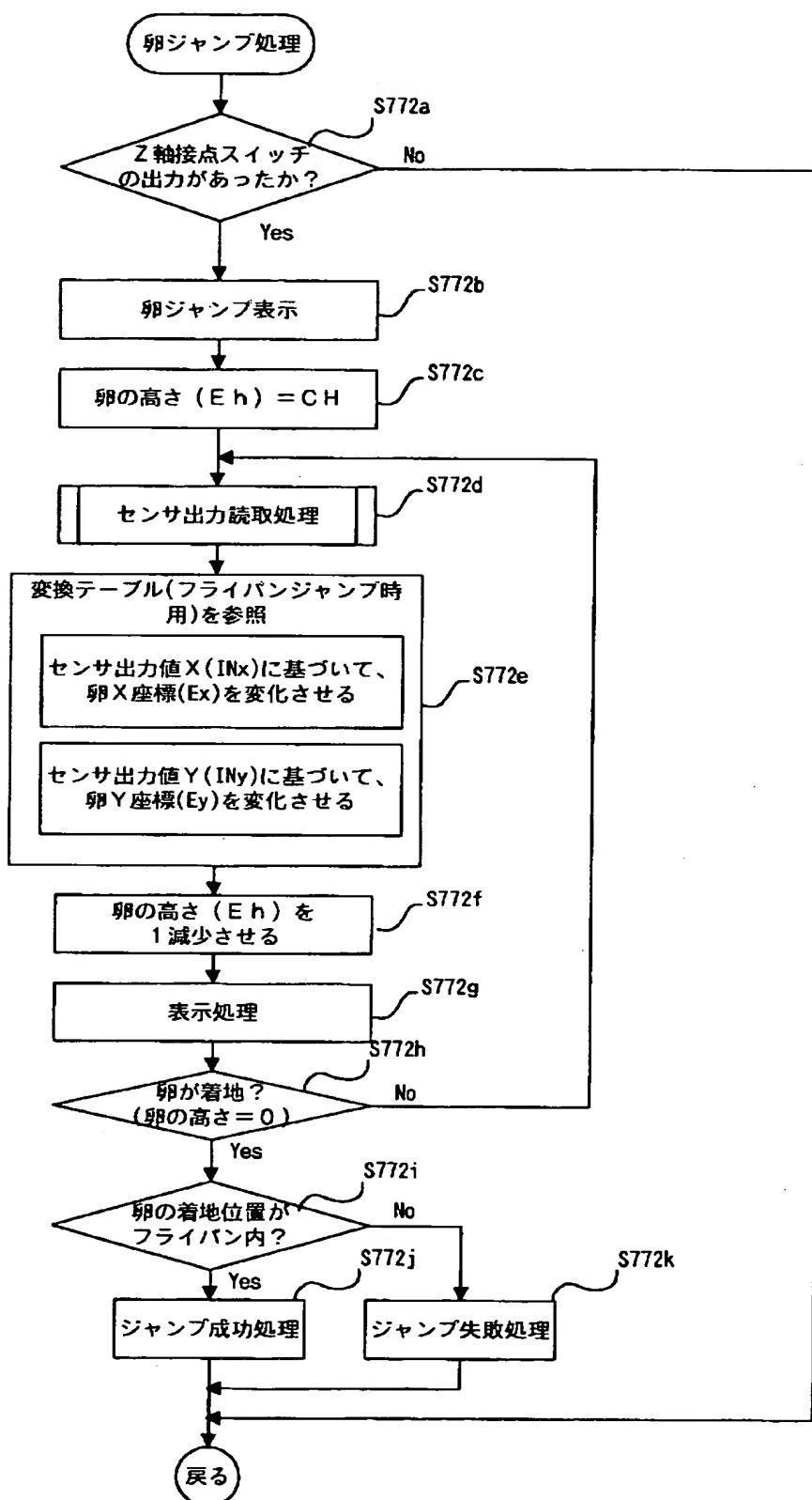
【図56】



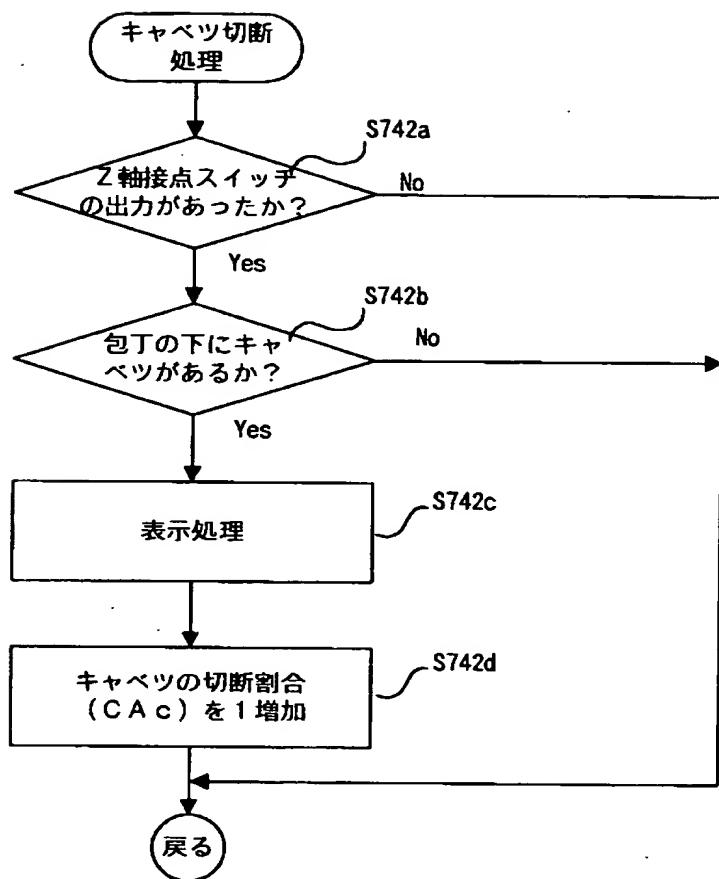
【図57】



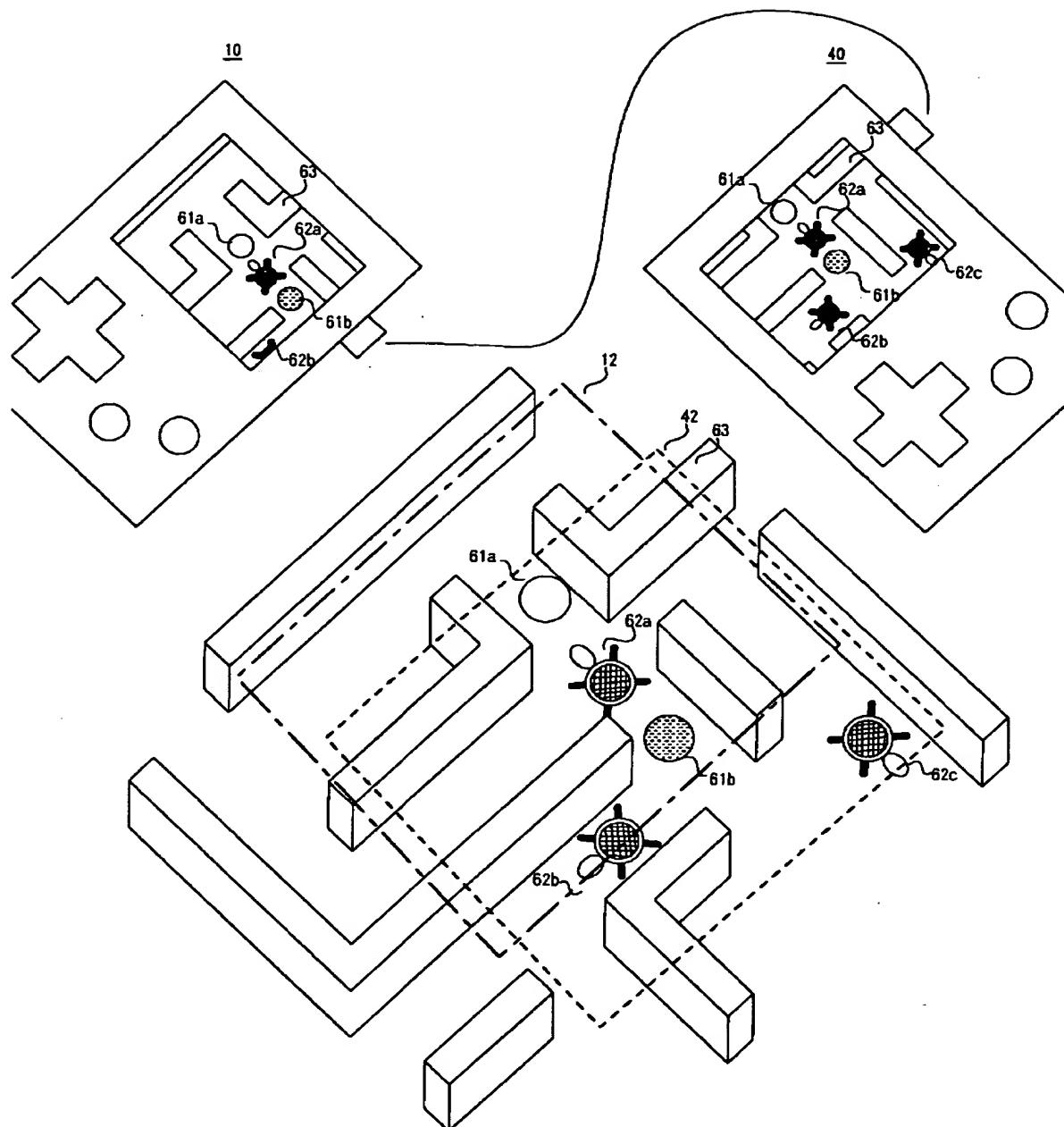
【図58】



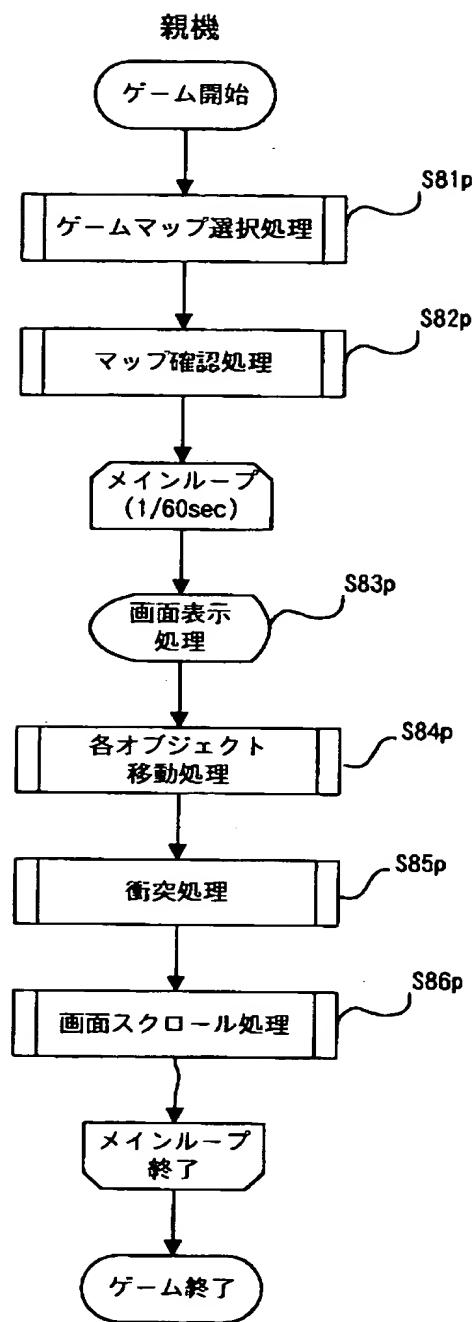
【図59】



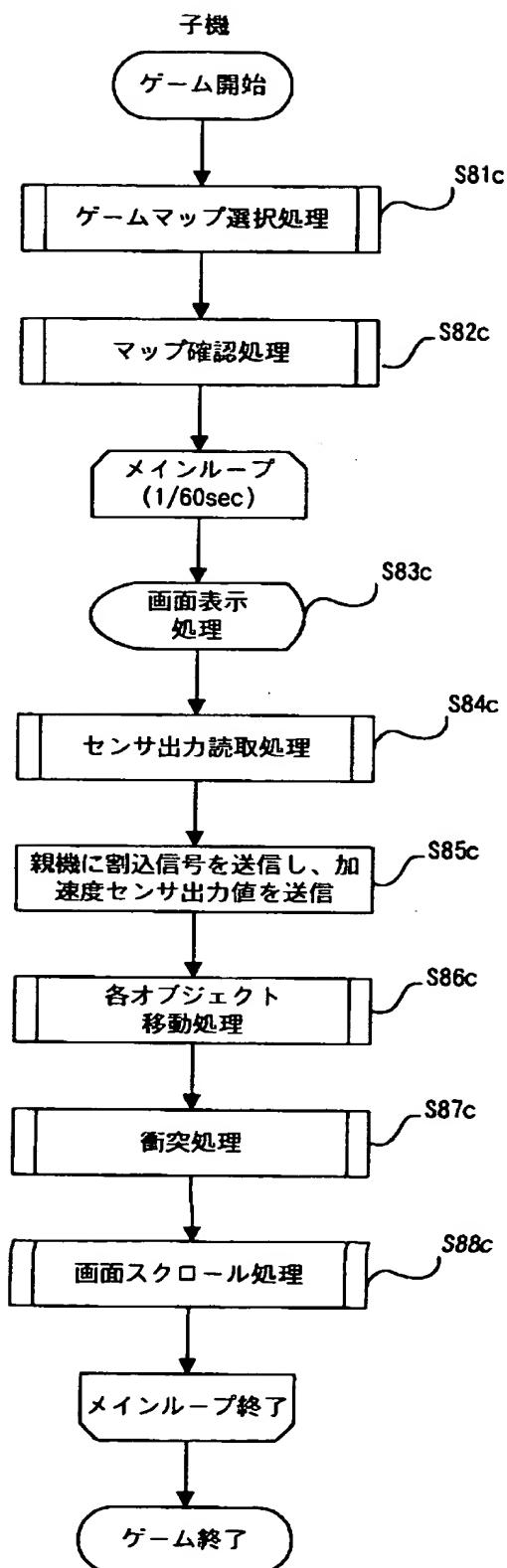
【図60】



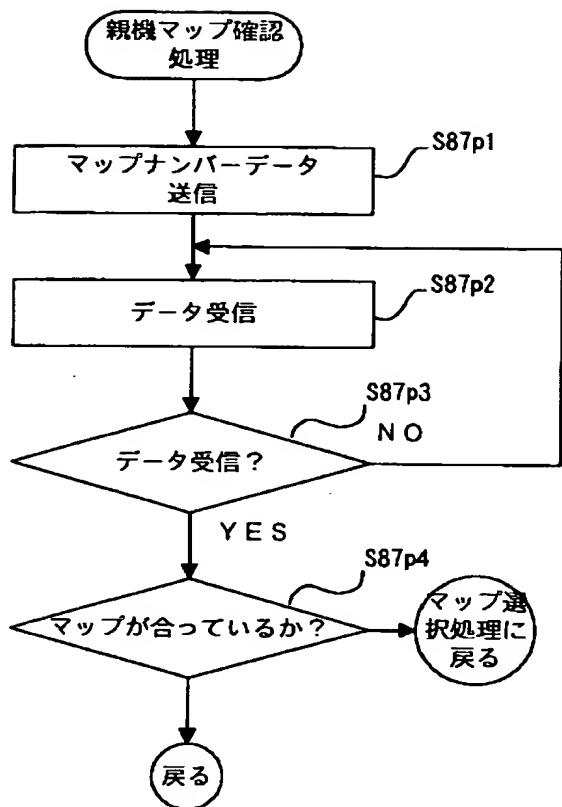
【図61】



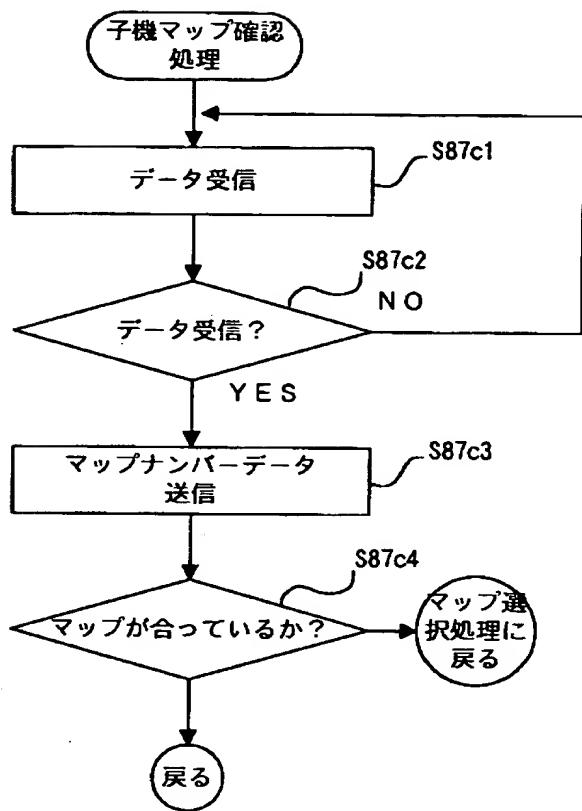
【図62】



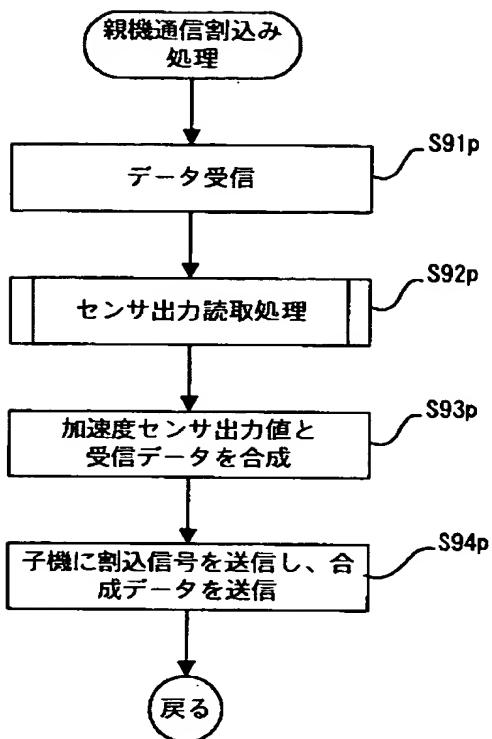
【図63】



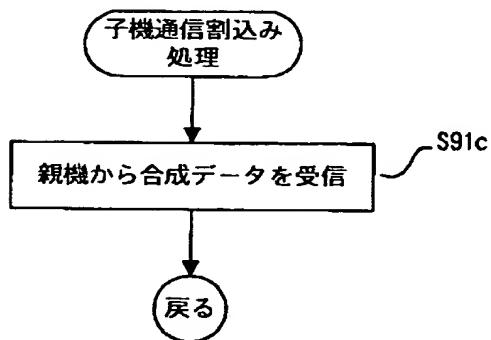
【図64】



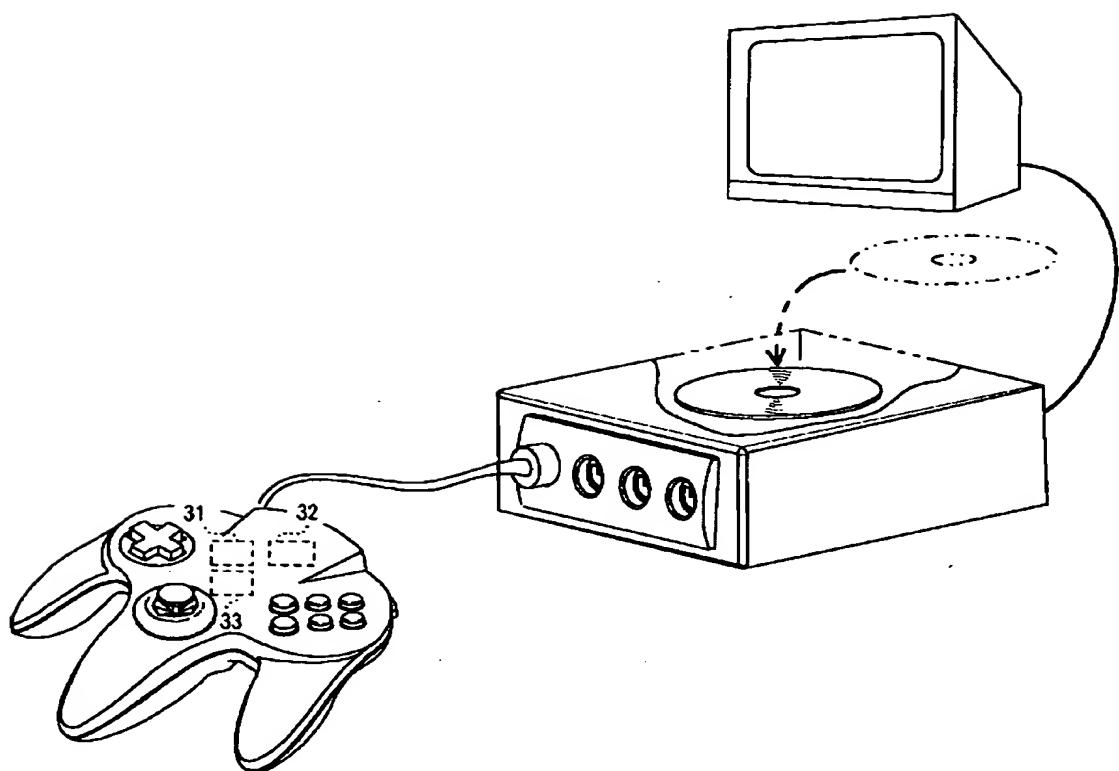
【図65】



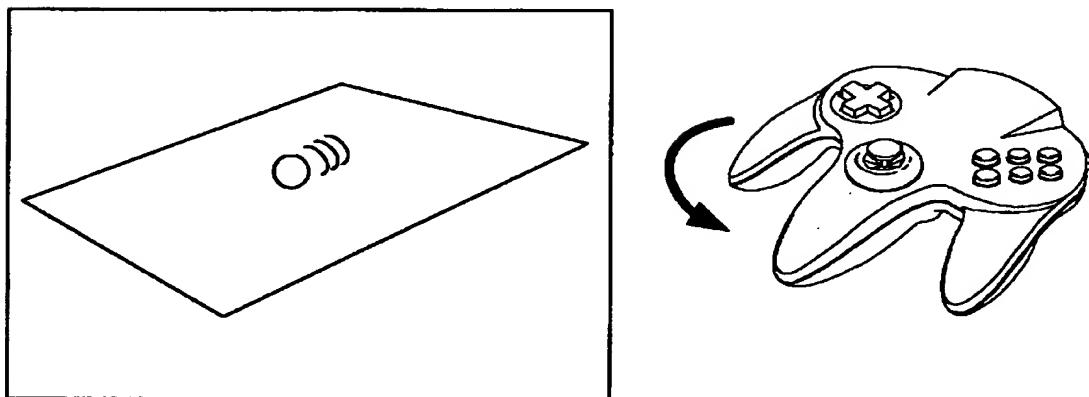
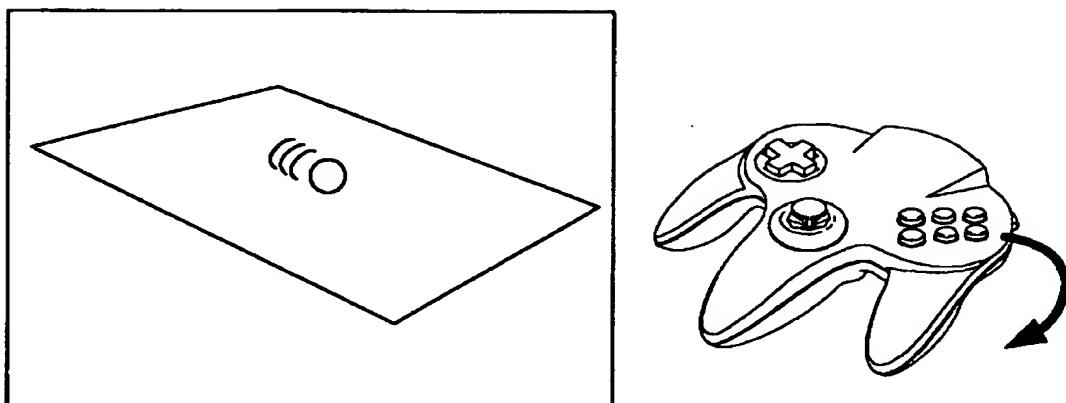
【図66】



【図67】



【図68】



【書類名】 要約書

【要約】

【課題】

簡単な操作でゲーム空間の状態を変化させることができ、プレイヤが操作方法に熟練する必要がなく、ゲームプレイに集中できるとともに、熱中度を高められる、ゲームシステム及びそれに用いられるゲーム情報記憶媒体を提供する。

【解決手段】

プレイヤによって把持されるハウジングに変化状態検出手段を備える。変化状態検出手段はハウジングに加えられた変化の量（例えば、傾き量、運動量、衝撃量等）と変化方向（例えば、傾き方向、運動方向、衝撃方向等）の少なくとも一方を検出する。シミュレーションプログラムが変化状態検出手段の出力に基づいて、ハウジングに加えられた変化量と変化方向の少なくとも一方に関連させて、ゲーム空間の状態を変化させるようにシミュレートする。

【選択図】 図3

認定・付加情報

特許出願の番号	特願2000-174573
受付番号	50000722676
書類名	特許願
担当官	渡辺 正幸 7072
作成日	平成12年 6月14日

<認定情報・付加情報>

【提出日】 平成12年 6月 9日

次頁無

出願人履歴情報

識別番号 [000233778]

1. 変更年月日 1990年 9月 6日

[変更理由] 新規登録

住 所 京都府京都市東山区福稻上高松町60番地  
氏 名 任天堂株式会社